Calculating response functions of coupled oscillators using quantum phase estimation

Sven Danz [1,2,*] Mario Berta [3,4] Stefan Schröder [5] Pascal Kienast,[5] Frank K. Wilhelm [1,2] and Alessandro Ciani [1]

[1]*Institute for Quantum Computing Analytics (PGI-12), Forschungszentrum Jülich, 52425 Jülich, Germany*

[2]*Theoretical Physics, Saarland University, 66123 Saarbrücken, Germany*

[3]*Institute for Quantum Information, RWTH Aachen University, 52074 Aachen, Germany*

[4]*Department of Computing, Imperial College London, London SW7 2RH, United Kingdom*

[5]*Fraunhofer Institute for Production Technology IPT, 52074 Aachen, Germany*

(Received 18 June 2024; accepted 23 May 2025; published 13 June 2025)

We study the problem of estimating frequency response functions of systems of coupled, classical harmonic oscillators using a quantum computer. The functional form of these response functions can be mapped to a corresponding eigenproblem of a Hermitian matrix H, thus suggesting the use of quantum phase estimation. Our proposed quantum algorithm operates in the standard s-sparse, oracle-based query access model. For a network of N oscillators with maximum norm $\|H\|_{\max}$, and when the eigenvalue tolerance ε is much smaller than the minimum eigenvalue gap, we use $\mathcal{O}(\log(Ns\|H\|_{\max}/\varepsilon)$ algorithmic qubits and obtain a rigorous worst-case query complexity upper bound $\mathcal{O}(s\|H\|_{\max}/(\delta^2\varepsilon))$ up to logarithmic factors, where δ denotes the desired precision on the coefficients appearing in the response functions. Crucially, our proposal does not suffer from the infamous state preparation bottleneck and can as such potentially achieve large quantum speedups compared to relevant classical methods. As a proof-of-principle of exponential quantum speedup, we show that a simple adaptation of our algorithm solves the random glued-trees problem in polynomial time. We discuss practical limitations as well as potential improvements for quantifying finite-size, end-to-end complexities for application to relevant instances.

DOI: 10.1103/5c6d-r1fb

I. INTRODUCTION

It is part of the experience of scientists and engineers that many relevant problems appearing in nature can be formulated as the problem of finding the eigenvalues and eigenvectors of a matrix. Consequently, a natural question to ask is whether quantum computers can provide some advantages in solving eigenproblems, compared to standard classical algorithms. The answer to this question is the subject of active research, but it is highly nontrivial and depends on the fine details of the problem itself.

In this paper, we focus on a particular category of eigenproblems, namely those associated with a system of N coupled harmonic oscillators. Needless to say, systems of coupled harmonic oscillators are ubiquitous in nature, as they can be used to effectively model the low energy dynamics of apparently very different physical systems, from electrical circuits to the vibration of molecules. In essence, our goal is to perform a modal analysis of the system, focusing on the calculation of response functions in frequency (or Laplace) domain. Response functions describe how much an external perturbation applied to a certain oscillator, influences the dynamics of another oscillator in the system. These functions have an analytical functional form that is related to an eigenproblem defined on the system of coupled oscillators. We thus focus on the question of how these response functions can be extracted using a quantum algorithm and in which cases we might expect significant quantum speedups.

A. Related work

The idea of using a quantum computer for eigenproblems dates back to the early days of quantum algorithm development [1]. In the quantum chemistry setting, where one is typically interested in characterizing the low-lying eigenvalues and eigenstates of a molecular Hamiltonian, it is in fact considered one of the most promising applications of quantum computers [2–4].

Broadly speaking, one can distinguish between two categories of quantum algorithms for eigenproblems, namely algorithms based on the variational quantum eigensolver [5,6], which lack rigorous scaling guarantees, and those based on the quantum phase estimation (QPE) subroutine [7–9]. In this work, we focus on the latter, and in particular on a formulation of standard QPE for eigenvalue estimation that relies on the block-encoding of the matrix of interest in a (larger) unitary [4,10–16]. A key feature of the problem that we exploit is that to estimate the response functions, we do not need to invoke a state preparation subroutine that prepares a desired eigenvector as a quantum state, for instance via controlled rotations [17], reject sampling method [18], eigenvalue transform [19] or adiabatic state preparation (see discussion in Ref. [4]). These kinds of state preparation subroutines hide additional complexity and can ruin potential

*Contact author: s.danz@fz-juelich.de

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

quantum speedups when properly taken into account. Instead, as we will show, in our case the omnipresent state preparation problem in QPE reduces to the preparation of a product state, from which we need to determine the relevant eigenvalues and associated weights via importance sampling. The simultaneous estimation of multiple eigenvalues using either QPE or other quantum subroutines has been studied in the literature using different approaches [20–23].

The works that are most related to ours on resolving coupled harmonic oscillators on a quantum computers are Refs. [24,25]. Reference [24] analyzes the problem of obtaining the classical time-evolution of a system of coupled harmonic oscillators. To do this, the authors employ a particular encoding of the positions and momenta of the oscillators in the amplitudes of an n-qubit quantum system. On this system, a Hamiltonian can be defined that mimics the time-evolution of the system of coupled oscillators at any time. With this approach, while one cannot efficiently access the full solution, it is possible to estimate efficiently relevant quantities such as the kinetic energy of a subset of oscillators. Moreover, Ref. [24] also shows that the random glued-trees problem originally studied in Ref. [26] can be mapped to a problem on coupled oscillators and solved efficiently using the quantum approach. We complement this result by showing that the glued-trees problem can also be solved efficiently with our QPE approach by a simple adaptation of the algorithm for the estimation of response functions. An interesting open question is whether response functions can also be estimated using the time-evolution approach of Ref. [24].

Reference [25] instead focuses on a normal mode analysis of systems of coupled oscillators, in the same spirit as our work, and essentially using the same kind of embedding of the problem in quantum systems. However, Ref. [25] does not discuss the application to the estimation of response functions and assumes that a good approximation of the eigenvectors can be prepared in a quantum system. While this would also allow to estimate response functions in principle, we show that it is not necessary for this task.

Finally, our work is also connected to quantum approaches to solve the wave equation [27] or more generally for finite element simulations of solid structures [28,29]. In fact, as we detail in the next section, once discretized, these problems can be mapped to a system of coupled harmonic oscillators.

B. Background on applications

The solution of ordinary and partial differential or integral equations describing technical and physical processes, such as the deformation of components under given loads or heat conduction in solids, is rarely successful analytically and must be solved numerically. The necessary transfer to a finite-dimensional equivalent problem requires a discretization process using methods such as the finite element or finite volume method [30]. In our work, we are particularly interested in problems for which this discretization step leads to an effective model of coupled harmonic oscillators. A prominent example of these problems are those that emerge in the manufacturing industry as we detail below.

Due to their time efficiency, finite element models are often used in practice to simulate machining processes, such as milling. Milling is a metal cutting manufacturing process

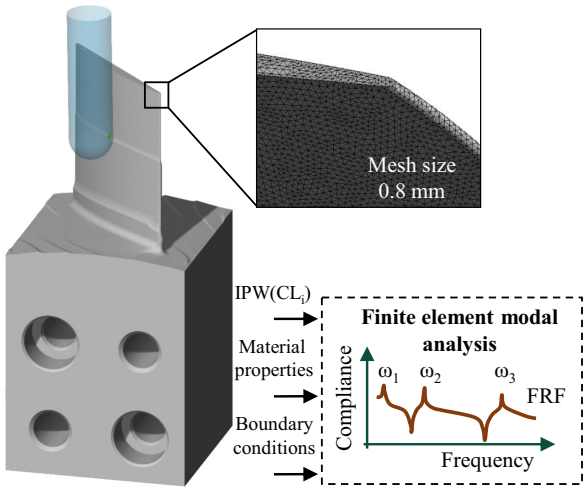


FIG. 1. In-process-workpiece (IPW) of a single blade demonstrator (left) and the detailed representation of the finite element mesh. The cutting tool is shown in blue. With the cutter location-dependent IPW, material properties (density, Young's modulus, Poisson's ratio), and boundary conditions as input, the finite element modal analysis can be conducted and the frequency response function (FRF) calculated by solving the corresponding eigenproblem. Typical solvers used for this task are based on the QR algorithm and have complexity $\mathcal{O}(N^3)$ [31–33].

that uses the circular cutting motion of a tool, usually with multiple cutting edges, to produce a vast variety of surfaces on a workpiece (see Fig. 1).In all milling processes, in contrast to other processes, such as turning and drilling for instance, the cutting edges are not constantly engaged. Still, at least one cutting interruption per cutting edge occurs with each revolution of the tool [34]. Due to the constant cutting interruptions, depending on the milling cutter speed, a dynamic excitation of the workpiece and the tool can occur, which can have a negative effect on the surface quality in the form of vibration marks. Dynamic process stability simulations are carried out to analyze the vibrations and to improve the process design for the milling of thin-walled components [35–39]. The ultimate goal of these simulations is to obtain response functions over a certain frequency range, i.e., the typical frequency range of the cutting tool, that provide information about the effect that the cutting tool at a certain location has on the workpiece at other, possibly different, locations. For this particular case, the response function is a so-called compliance (with units m/N) as a function of the frequency of the cutting edge, as shown in Fig. 1, which effectively quantifies the vibration of the workpiece caused by the cutting tool.

Reference [40] analyzed a workflow for milling dynamics simulation, examining computational time and identifying numerical problems that can be suited for quantum algorithms. The milling dynamics workflow considers the dynamic behavior of the workpiece, which continuously varies due to changes in the stiffness and mass of the workpiece caused by the material removal and the position-dependent force excitation by the milling tool [41]. This analysis highlighted the fact that finite element modal analysis is typically a computationally intensive simulation application in milling of

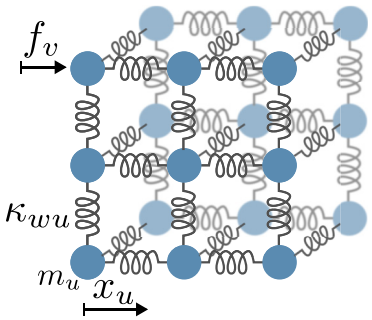


FIG. 2. Representation of coupled oscillators. Pointlike masses m_u in blue are connected by springs with spring constant κ_{wu}. The force f_v applied to mass v yields a displacement x_u at mass u.

thin-walled aerospace components, and as such could benefit from possible advantages that quantum algorithms could offer for this task.

C. Structure

The rest of the paper is structured as follows. In Sec. II, we discuss the connection between the computation of the response function of coupled oscillators and the solution of an eigenproblem. We further describe briefly our quantum algorithm and its scaling behavior, which is summarized in Theorem 1. We finish the section with a critical assessment of the limitations and possible improvements of our quantum approach. In Sec. III, we describe our proposed quantum algorithm in more detail, for which a pseudocode is provided in Algorithm 1. Section IV discusses how the random-glued trees problem can be solved using the QPE subroutine. We then summarize our results and provide an outlook in Sec. V. The Appendixes contain a more in-depth derivations of the analytical form of the response function (Appendix A), descriptions of the quantum subroutines used in the algorithm (Appendix B), a detailed error analysis (Appendix C), a modified version of the algorithm for nonlocal response functions (Appendix D), and a scaling analysis of the algorithm applied to the random glued-trees problem (Appendix E).

II. PROBLEM STATEMENT AND MAIN RESULTS

A. Linear response functions as an eigenproblem

Let us consider a system of masses coupled by spring constants as shown in Fig. 2 and let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph with $\mathcal{V}$ the set of vertices and $\mathcal{E}$ the set of edges, that describes the network of oscillators. With each vertex $u \in \mathcal{V}$ we associate a harmonic oscillator with mass $m_u > 0$ and a coupling to a common wall with spring constant $\kappa_u > 0$. We denote by $N = |\mathcal{V}|$ the number of oscillators. With each edge $(u, v) \in \mathcal{E}$ in the graph, instead, we associate a coupling spring constant $\kappa_{uv} > 0$. In what follows, we implicitly assume an ordering of the vertices, i.e., we associate with each $u \in \mathcal{V}$ a number between 1 and N. For notational simplicity, we also denote this number by u, since its meaning is clear from the context. Additionally, we denote by $N_{\mathcal{G}}[u]$ the closed neighborhood of $u \in \mathcal{V}$, i.e., the set of vertices connected to u by an edge plus u itself. We call $s_u = |N_{\mathcal{G}}[u]|$ and define the sparsity s as

ALGORITHM 1. Calculation of the local response function $G_{uu}(v)$ on a quantum computer. The first part estimates classically the register and sample size required by the quantum circuit following in the loop over t. The classical postprocessing in the end is required to extract the response function from the measurements.

Data:

- Oscillator u
- Mass m_u
- Renormalization $s\|H\|_{\max}$
- Minimum eigenvalue gap $\Delta_\lambda^{(u)}$
- Tolerances ε, δ, ζ
- Controlled quantum walk operator cV which contains the oracles O_ϑ and O_P
- Number of oscillators N
- Number of distinguishable eigenvalues N_u

Result: Local response function $G_{uu}(v)$

begin

m, n, $a+l \leftarrow$ compute the required register sizes
$N_S \leftarrow$ compute the required sample size
for $t \in \{1, \ldots, N_S\}$ **do**
  $\mathtt{reg}_{\text{phase}} \leftarrow \sum_{i=0}^{2^m-1} |i\rangle$
  $\mathtt{reg}_{\text{state}} \leftarrow |u\rangle$
  $\mathtt{reg}_{\text{ancilla}} \leftarrow |0\rangle^{\otimes a+l}$
  for $i \in \{0, \ldots, m-1\}$ **do**
    cV^{2^i} to $[\mathtt{reg}^{(i)}_{\text{phase}}, \mathtt{reg}_{\text{state}}, \mathtt{reg}_{\text{ancilla}}]$
  QFT† to $\mathtt{reg}_{\text{phase}}$
  $\varphi_t \leftarrow$ measure $\mathtt{reg}_{\text{phase}}$
define $P(\varphi) = \sum_{t=1}^{N_S} \delta(\varphi - \varphi_t)/N_S$
$\boldsymbol{\varphi} \leftarrow$ identify N_u peaks in $P(\varphi)$
for $j \in \{j : \varphi_j \in \boldsymbol{\varphi}\}$ **do**
  $\lambda_j \leftarrow s\|H\|_{\max} \cos(2^{1-m}\pi\varphi_j)$
  $W_{uj}^2 \leftarrow 2\sum_{q=-Q}^{Q-1} P(\lceil \varphi_j \rceil + q)$
$G_{uu}(v) \leftarrow m_u^{-1} \sum_{j=1}^{N_u} W_{uj}^2/(\lambda_j + v^2)$

$s = \max_{u\in\mathcal{V}} s_u$. For later convenience, we also define the set of vertices $\tilde{N}_{\mathcal{G}}[u]$ such that $N_{\mathcal{G}}[u] \subset \tilde{N}_{\mathcal{G}}[u]$ and $|\tilde{N}_{\mathcal{G}}[u]| = s$. Basically, $\tilde{N}_{\mathcal{G}}[u]$ is the neighborhood of u enlarged with dummy vertices to ensure that the total number of elements in the set is s.

1. Unperturbed dynamics

We can compactly write the dynamical equations that govern the motion of the coupled oscillator systems, by defining two $N \times N$ matrices, namely a diagonal mass matrix $\boldsymbol{M}$ with matrix elements

$$M_{uv} = m_u \delta_{uv}, \tag{1}$$

and a stiffness matrix $\boldsymbol{K}$ with matrix elements

$$K_{uv} = \begin{cases} \kappa_u + \sum_{w\in N_{\mathcal{G}}[u]\setminus\{u\}} \kappa_{wu} & \text{if} \quad u = v, \\ -\kappa_{uv} & \text{if} \quad (u, v) \in \mathcal{E}, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Additionally, we denote by $\boldsymbol{x}$ the N-dimensional column vector whose elements are the oscillator positions x_u with respect to the equilibrium state.

The system evolves according to the dynamical equation

$$\boldsymbol{M}\frac{d^2\boldsymbol{x}}{dt^2} = -\boldsymbol{K}\boldsymbol{x}. \tag{3}$$

To transform this problem into an eigenproblem, let us first define the matrix

$$\boldsymbol{H} = \boldsymbol{M}^{-1/2}\boldsymbol{K}\boldsymbol{M}^{-1/2}, \tag{4}$$

which, as we will see, will play the role of a Hamiltonian in our quantum algorithm. Note that since $\boldsymbol{M}$ is positive definite, also $\sqrt{\boldsymbol{M}}$ is positive definite and thus invertible. The matrix $\boldsymbol{H}$ is Hermitian and positive-semidefinite and we denote its eigenvalues by $\lambda_j \geqslant 0$ for $j \in \{1, \ldots, N\}$. We now show that solving the dynamical equation Eq. (3) is equivalent to diagonalizing the matrix $\boldsymbol{H}$. We denote by $\boldsymbol{W}$ an orthogonal matrix that diagonalizes $\boldsymbol{H}$,

$$\boldsymbol{\Lambda} = \boldsymbol{W}^T\boldsymbol{H}\boldsymbol{W}, \tag{5}$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix with the eigenvalues of $\boldsymbol{H}$ on the diagonal, i.e., $\Lambda_{jj} = \lambda_j$. Defining the vector of normal mode variables $\boldsymbol{y}$ as

$$\boldsymbol{y} = \boldsymbol{W}^T\boldsymbol{M}^{1/2}\boldsymbol{x}, \tag{6}$$

we get an uncoupled system of harmonic oscillators satisfying

$$\frac{d^2\boldsymbol{y}}{dt^2} = -\boldsymbol{\Lambda}\boldsymbol{y} \Rightarrow \frac{d^2y_j}{dt^2} = -\lambda_j y_j. \tag{7}$$

Since $\lambda_j \geqslant 0$ they are usually denoted as $\lambda_j = \omega_j^2$, with ω_j the resonance frequency of the normal mode j. Note that Eq. (6) implies that

$$x_u = \frac{1}{\sqrt{m_u}}\sum_{j=1}^{N} W_{uj}y_j. \tag{8}$$

2. Adding an external force

Let us now add an external force to Eq. (3). In particular, we assume that a time-dependent force $f_v(t)$ is applied to the oscillator $v \in \mathcal{V}$ and we gather all the forces in an N-dimensional column vector $\boldsymbol{f}(t)$. Including the forcing term, the dynamical equation reads

$$\boldsymbol{M}\frac{d^2\boldsymbol{x}}{dt^2} = -\boldsymbol{K}\boldsymbol{x} + \boldsymbol{f}. \tag{9}$$

Equation (9) gives rise to a convolution relation between $\boldsymbol{f}(t)$ and $\boldsymbol{x}(t)$,

$$\boldsymbol{x}(t) = \int_{-\infty}^{+\infty} d\tau \boldsymbol{g}(t-\tau)\boldsymbol{f}(\tau), \tag{10}$$

where $\boldsymbol{g}(t)$ is an $N \times N$ matrix that we call the matrix response function. The matrix elements $g_{uv}(t)$ can simply be interpreted as the response of the oscillator u when a forcing term on the oscillator v, $f_v(t)$, is a Dirac delta function, i.e., $f_v(t) = \delta(t)$, while no force is applied to the other oscillators.[1] This interpretation also suggests that causality implies $\boldsymbol{g}(t) = 0$ if $t < 0$ (i.e., we consider a retarded response function). Thus, we can simply take the upper limit of the integral in Eq. (10) to be t rather than $+\infty$. Equation (10) is the defining equation of a linear time-invariant system [42].

Now, let us introduce the bilateral Laplace transform of a generic function $\boldsymbol{z}(t)$ as

$$\boldsymbol{Z}(\upsilon) = \int_{-\infty}^{+\infty} dt e^{-\upsilon t}\boldsymbol{z}(t), \quad \upsilon = \sigma + i\omega \in \mathbb{C}. \tag{11}$$

The Laplace transform is defined only in the region of convergence, that is the complex domain where the integral in Eq. (11) converges. From the convolution theorem for the Laplace transform Eq. (10) in Laplace domain becomes

$$\boldsymbol{X}(\upsilon) = \boldsymbol{G}(\upsilon)\boldsymbol{F}(\upsilon), \tag{12}$$

with $\boldsymbol{G}(\upsilon)$ the Laplace transform of the response function in time domain $\boldsymbol{g}(t)$, while $\boldsymbol{X}(\upsilon)$ and $\boldsymbol{F}(\upsilon)$ the Laplace transform of the position vector $\boldsymbol{x}(t)$ and the vector of forces $\boldsymbol{f}(t)$, respectively.

In the main text, we focus on a quantum algorithm to determine the diagonal elements of $\boldsymbol{G}(\upsilon)$ that take the form

$$G_{uu}(\upsilon) = \frac{1}{m_u}\sum_{j=1}^{N}\frac{W_{uj}^2}{\lambda_j + \upsilon^2}, \tag{13}$$

which we call the local response functions. However, it is also possible to obtain the off-diagonal elements of $\boldsymbol{G}(\upsilon)$ or nonlocal response function at the price of adding a Hadamard test, as we show in Appendix D. We provide a derivation of Eq. (13), as well as the more general definition for local and nonlocal response, in Appendix A.

Equation (13) shows that the response function $G_{uu}(\upsilon)$ is completely determined once we solve the eigenproblem associated with $\boldsymbol{H}$. In fact, all we need to compute the response function is the eigenvalues λ_j and the coefficients W_{uj}^2, with W_{uj} the matrix elements of the orthogonal matrix $\boldsymbol{W}$ that diagonalizes $\boldsymbol{H}$. As discussed in Sec. I B, the response function is the relevant quantity to study in practical applications. State-of-the-art classical algorithms that are used in practice for the eigenproblem at hand, such as the QR algorithm, need at least $\mathcal{O}(N^3)$ arithmetic operations. In fact, most of the fastest classical algorithms for eigenvalue computation rely on the Householder transformation which transforms an $N \times N$ matrix into the Hessenberg form (almost triangular). This transformation needs $4N^3/3$ operations (see Chap. 11, p. 474 in Ref. [32]) and dominates the scaling of the QR algorithm. The sparsity of $\boldsymbol{H}$ reduces the runtime to $\mathcal{O}(N^2 s)$.

The quantum algorithm proposed in this manuscript is designed to compute all eigenpairs of a sparse matrix. However, we will give a bief outlook on the scaling of advanced classical eigenpair solvers that compute only a small subset of all eigenpairs. Both the power and the more advanced Lanczos method [43,44] allows us to approximate the k largest eigenvalues of $\boldsymbol{H}$ in $\mathcal{O}(Nk(s+k))$ arithmetic operations if $\boldsymbol{H}$ has well-separated eigenpairs [45,46]. Combined with the inverse power method and a shifted matrix, this yields an approximation to an eigenvalue close to any chosen point [47]. However, the response function computed with those methods are only accurate at extremes of the full spectrum. Its runtime

[1] The response function is also called the Green's function for the system of coupled oscillators.

scales worse for matrices with small eigenvalue gaps and its memory requirements with $\mathcal{O}(N(s+k))$ storage registers is also not small [45].

In the following quantum treatment, we refer to the $\boldsymbol{H}$ matrix in Eq. (4) as the Hamiltonian. From now on we avoid using the bold symbols, simply denoting the Hamiltonian as H, and do the same for the other matrices.

B. Main idea

In this work, we describe a quantum algorithm that computes the local and nonlocal response functions of a system of coupled oscillators. We start by providing the basic intuition behind the algorithm for the case of the local response function in Eq. (13), while we refer to Sec. III and the Appendixes for more technical details. To compute $G_{uu}(\upsilon)$, we need to determine the eigenvalues λ_j and the coefficients W_{uj}^2. In our approach, each vertex $u \in \mathcal{V}$ is associated with a bitstring $|u\rangle$ of length n in the computational basis, where the number of qubits is $n = \lceil \log_2 N \rceil$. For simplicity, from now on we assume that N is a power of two, i.e., $N = 2^n$. Since, W is the orthogonal transformation that diagonalizes H, we can express the bitstring $|u\rangle$ as a linear combination of the eigenstates $|\lambda_j\rangle$ of H:

$$|u\rangle = \sum_{j=1}^{N} W_{uj}|\lambda_j\rangle. \tag{14}$$

Thus, we can interpret W_{uj} as weights in the previous decomposition. Equations (13) and (14) convey the core idea behind the algorithm. If we prepare the bitstring $|u\rangle$ and perform measurement in the eigenbasis $|\lambda_j\rangle$ of H, then we will get the eigenvalues λ_j with probability W_{uj}^2. By repeating this procedure many times we can estimate the coefficients W_{uj}^2 via statistical averages and obtain an estimate for the response function $G_{uu}(\upsilon)$. Note that there is an implicit importance sampling in this procedure: the eigenstates that contribute the most to $G_{uu}(\upsilon)$ are also those whose eigenvalue will be measured more often. To implement the measurement in the eigenbasis we use QPE. The full algorithm is described in detail in Sec. III.

For the implementation of the quantum algorithm, we assume that we have oracle access to the matrix elements of H and that H is s-sparse, i.e., that each row has at most s nonzero elements. In particular, we assume that we have access to two matrix oracles. The first one is the position oracle O_P, that acts on the $2n$-qubit state $|u\rangle|0\rangle^{\otimes n}$ as

$$O_P|u\rangle|0\rangle^{\otimes n} = \frac{1}{\sqrt{s}} \sum_{v \in \tilde{N}_{\mathcal{G}}[u]} |u\rangle|v\rangle. \tag{15}$$

O_P stores in the second register a linear superposition of all the column indices that correspond to nonzero matrix elements at row u and the additional dummy indices in $\tilde{N}_{\mathcal{G}}[u]$ if necessary.

The second oracle is the angle oracle O_ϑ which contains information about the matrix elements of H. It encodes the angle

$$\vartheta_{uv} = \arccos \sqrt{\frac{|H_{uv}|}{\|H\|_{\max}}}, \tag{16}$$

and the sign of H_{uv} in an $(r+1)$-qubit ancilla register

$$O_\vartheta|u, v\rangle|0\rangle^{\otimes r+1} = |u, v\rangle|\vartheta_{uv}, \Theta(-H_{uv})\rangle, \tag{17}$$

with $\Theta(\cdot)$ the Heaviside step function and r the number of bits used to represent ϑ_{uv}. We renormalize H_{uv} in Eq. (16) with $\|H\|_{\max} = \max_{u,v}|H_{uv}|$. Specifically, since $0 \leqslant \vartheta_{uv} \leqslant \pi/2$, we assume the following r-bit representation of ϑ_{uv},

$$\vartheta_{uv} = \frac{\pi}{2} \sum_{l=1}^{r} \vartheta_{uv}^{(l)} 2^{-l}, \tag{18}$$

with $\vartheta_{uv}^{(l)} \in \{0, 1\}$. Thus, when we write $|\vartheta_{uv}\rangle$ in an r-qubit register, this means

$$|\vartheta_{uv}\rangle = \left|\vartheta_{uv}^{(1)}\right\rangle \left|\vartheta_{uv}^{(2)}\right\rangle \ldots \left|\vartheta_{uv}^{(r)}\right\rangle. \tag{19}$$

Finally, we assume that we have a quantum adder ADD available, which adds the information in one quantum register to another, defined as

$$\mathrm{ADD}|a\rangle|b\rangle = |a\rangle|b + a \mod 2^n\rangle, \tag{20}$$

where n is the number of qubits in the second register.[2] The inverse of this operation is a quantum subtractor $\mathrm{ADD}^\dagger = \mathrm{SUB}$. Circuit realizations of the quantum adder are well known in the literature [48–51]. Together with the matrix oracles O_P and O_ϑ, the quantum adder ADD is needed for the implementation of the block-encoding described in detail in Appendix B (see Fig. 5 in particular) and used in Sec. III.

The computational cost of the quantum adder will be neglected in the following query complexity analysis. This is because the implementation of the oracle O_ϑ usually requires multiple queries to the quantum adder. Hence, the additional use of one ADD per oracle is negligible in terms of required qubit number and computation time.

C. Complexity analysis

We now summarize the main results in terms of scaling with respect to relevant parameters. We start by defining the additional parameters that enter in the scaling analysis. First, we define the minimum, positive gap in the eigenvalue spectrum as

$$\Delta_\lambda = \min\{|\lambda_i - \lambda_j| : i, j \in \{1, \ldots, N\},\ |\lambda_i - \lambda_j| > 0\}. \tag{21}$$

Additionally, we define the minimum, positive eigenvalue gap at oscillator u as

$$\begin{aligned}\Delta_\lambda^{(u)} = \min\{&|\lambda_i - \lambda_j| : i, j \in \{1, \ldots, N\},\\ &|\lambda_i - \lambda_j| > 0, W_{ui}^2 > 0, W_{uj}^2 > 0\},\end{aligned} \tag{22}$$

with W_{ui} and W_{uj} being the coefficients in Eq. (14). This is the parameter that matters in our algorithm since, in the presence of degeneracies, weights associated with the same eigenvalue would just be gathered together in Eq. (13) and eigenvectors that have no support at oscillator u give zero contribution. Obviously, $\Delta_\lambda^{(u)} \geqslant \Delta_\lambda$. The scaling further depends on the required, additive tolerances ε and δ for the eigenvalues λ_j

[2]The number of qubits in the first register does not need to be equal to the one in the second register.

TABLE I. Parameters used in the complexity analysis.

Symbol	Meaning
N	Number of rows or columns of H
s	Sparsity of H
$\|H\|_{\max}$	Maximum absolute value of the entries of H
$\Delta_\lambda^{(u)}$	Minimum eigenvalue gap at oscillator u
ε	Additive eigenvalue tolerance
δ	Additive weight tolerance
ζ	Probability of failure
N_u	Number of nonzero W_{uj} at oscillator u

and weights W_{uj}^2, respectively, which we treat as independent parameters of our choice.[3] Moreover, we define the parameter N_u to be the number of eigenvectors that have support at oscillator u, that is for which $W_{uj}^2 > 0$. Note that in the worst-case scenario $N_u = N$ from the true value. Finally, we denote by ζ the probability that the estimate of W_{uj}^2 deviates by more than δ. All the parameters used in the complexity analysis are summarized in Table I.

Our main result is described by the following theorem.

Theorem 1. Let H be a $N \times N$ Hermitian matrix that describes a system of coupled oscillators as in Eq. (4) and let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be the graph associated with the network of oscillators. Let H be s-sparse with maximum norm $\|H\|_{\max}$. Given an oscillator $u \in \mathcal{V}$, there exists a quantum algorithm that approximates the eigenvalues λ_j of H and the weights W_{uj}^2 in Eq. (13), with additive error ε and δ, respectively, that requires

$$n_{\text{tot}} = \mathcal{O}\left(\log_2\left(Ns\|H\|_{\max}\max\left(\frac{1}{\varepsilon}, \frac{1}{\delta\Delta_\lambda^{(u)}}\right)\right)\right) \tag{23}$$

qubits and

$$N_{\text{queries}}^{(\text{tot})} = \mathcal{O}\left(\frac{s\|H\|_{\max}}{\delta^2}\ln\left(\frac{N_u}{\zeta}\right)\max\left(\frac{1}{\varepsilon}, \frac{1}{\delta\Delta_\lambda^{(u)}}\right)\right) \tag{24}$$

total queries of the matrix oracles O_P and O_ϑ in Eqs. (15) and (17), respectively.

A detailed error analysis of the QPE-based algorithm behind Theorem 1 is presented in Appendix C. In what follows, we provide further comments on the complexity of our result.

First, we note that Eq. (23) implies that ε should be taken at least $\varepsilon < \Delta_\lambda^{(u)}$, since $\delta < 1$, to use the full potential of n_{tot} qubits. Moreover, in Eq. (23), we are not considering the ancilla qubits necessary to implement the oracles O_P, O_ϑ, since these are problem-dependent. If we use r bits to represent the matrix elements, then the required number of bits in classical algorithms is usually $\mathcal{O}(rN^2)$, while for s-sparse matrices it can be $\mathcal{O}(rsN)$ in the best case. The gate complexity of our quantum algorithm depends on the complexity of implementing the oracles O_P and O_ϑ, which are, as mentioned above, problem-dependent. Generally, as long as the matrix elements of a matrix can be accessed efficiently classically, this should also hold on a quantum computer, although at the price of having several ancillas to make the computation reversible. As an example, for the case of a linear chain of oscillators, we provide a full compilation of the oracles into elementary gates in the code available in the repository associated with this manuscript (see the "Data Availability" section). As such, we choose to present the complexity of our algorithm in terms of the number of oracle queries N_{queries}. The query complexity of a single run of QPE is (see Appendix C 1 c)

$$N_{\text{queries}} = \mathcal{O}\left(s\|H\|_{\max}\max\left(\frac{1}{\varepsilon}, \frac{1}{\delta\Delta_\lambda^{(u)}}\right)\right). \tag{25}$$

This is mostly due to the repetition of the controlled V operator in the core of the phase estimation (cf. Fig. 3 and Algorithm 1).

QPE is a probabilistic algorithm from which we sample multiple times to estimate the eigenvalues and the associated probabilities. As we show in Appendix C 2, the number of samples N_S needed to estimate each weight W_{uj}^2 via empirical averages with error δ and failure probability ζ scales as

$$N_S = \mathcal{O}\left(\frac{1}{\delta^2}\ln\left(\frac{N_u}{\zeta}\right)\right). \tag{26}$$

In the worst-case scenario when $N_u = N = 2^n$, the scaling would be logarithmic with the size of the matrix N, i.e., linear with the number of qubits n. We highlight the fact that this is not equivalent to requiring that the probability distribution W_{uj}^2 is estimated with error δ in total variation distance. Using the definition of total variation distance and Hoeffding's inequality similarly to Appendix C 2, one can show that in this case the scaling would be linear, and not logarithmic, with N_u. Moreover, we are not requiring to find all the N_u eigenvalues that contribute in the response function at oscillator u. This would require a number of samples scaling at least as N_u, but what matters for us is the evaluation of the weights W_{uj}^2 with error δ, since the weights appear at the numerator in Eq. (13). This leads to the inverse quadratic scaling in δ in Eq. (26).

Combining Eqs. (25) and (26), we obtain the total number of queries as in Eq. (24).

D. Critical assessment

1. Limitations of the algorithm

Neglecting δ for a moment, the computational cost [cf. Eqs. (23) and (24)] contains the inverse of

$$\frac{\min\left(\varepsilon, \Delta_\lambda^{(u)}\right)}{s\|H\|_{\max}}. \tag{27}$$

Assuming the worst-case scenario in which, $\Delta_\lambda^{(u)} = \Delta_\lambda$, and we want to guaranty a separation of all eigenvalues, the desired tolerance ε is required to satisfy $\varepsilon < \Delta_\lambda$. We further know that the 1-norm $\|H\|_1 = \max_v \sum_u |H_{uv}|$ is smaller than $s\|H\|_{\max}$. Thus, we have

$$\frac{\min\left(\varepsilon, \Delta_\lambda\right)}{s\|H\|_{\max}} \leqslant \frac{\Delta_\lambda}{s\|H\|_{\max}} \leqslant \frac{\Delta_\lambda}{\|H\|_1} \leqslant \frac{\Delta_\lambda}{\lambda_{\max}}, \tag{28}$$

where we used the upper limit for the maximum eigenvalue $\lambda_{\max} \leqslant \|H\|_1$. Hence, the number of oracle queries scales

[3]The smaller the error on the eigenvalues, the smaller the error on the weights, and vice versa. However, for our analysis it is easier to think about them as independent tolerances.

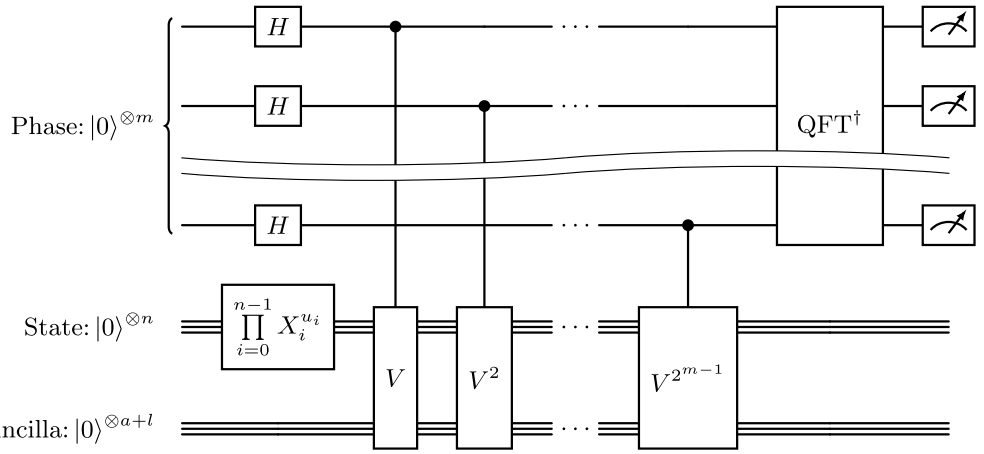


FIG. 3. QPE circuit for the computation of the local response function. The state preparation is realized by a series of NOT gates. QPE is performed using the walk operator V defined in Eq. (36) followed by the inverse quantum Fourier transform ($\mathrm{QFT}^\dagger$) [7]. Sampling from the phase register and repeating many times allows us to estimate the relevant eigenvalues λ_j and the weights W_{uj}^2 in Eq. (13).

with the ratio between the smallest eigenvalue gap and the maximum eigenvalue. This, in the general case, increases polynomially with the number of eigenvalues that have to fit between 0 and $\lambda_{\max}$, and accordingly it increases polynomially with the number of oscillators N.

The kinds of problems for which we can expect a polylogarithmic scaling in N of the total query complexity necessarily need to satisfy $\lambda_{\max}/\Delta_\lambda^{(u)} = \mathcal{O}(\mathrm{polylog}(N))$ in our rigorous worst-case scaling analysis. On the one hand, the random glued-trees problem we discuss in Sec. IV has this property, which ultimately originates from its very large degeneracy. This shows that there exists well-defined problems for which a large exponential speedup is possible. However, the task there is not the determination of response functions. On the other hand, if we simply consider a 1D periodic chain with N oscillators and unit spring constants and masses, then the eigenvalues can be obtained analytically and are given by [52]

$$\lambda_j = 2\left(1 - \cos\left(\frac{2\pi(j-1)}{N}\right)\right),$$

for $j \in \{1, \ldots, N\}$. For this problem $s\|H\|_{\max}$ is bounded, namely $s\|H\|_{\max} = 6$, but for large N the gap for this problem scales as $\Delta_\lambda = \mathcal{O}(N^{-2})$, yielding eventually a polynomial scaling of the number of queries with N. Notice that these considerations have no influence on the scaling with the tolerance on the weights δ, which is $1/\delta^3$ if we take it as an independent parameter [see Eq. (24)].

2. Possible improvements

Possible improvements in the scaling in the δ parameter could be achieved combining our algorithm with the quantum amplitude estimation subroutine [53]. Additionally, in the previous discussion, we have not restricted the domain of interest of the eigenvalues that are in general between 0 and $\lambda_{\max}$. However, in practice we might be interested in evaluating the response function only in a certain frequency range for which only eigenvalues $\lambda_j \in [\lambda_a, \lambda_b]$ matter. In fact, we might know in advance the typical frequency spectrum of the forcing term, as it happens in the milling case discussed in Sec. I B. In this case, we can approximate the response function as

$$G_{uu}(\upsilon) \approx \frac{1}{m_u} \sum_{j \in \mathcal{J}} \frac{W_{uj}^2}{\lambda_j + \upsilon^2}, \tag{29}$$

for $\mathrm{Im}(\upsilon) \in [\lambda_a, \lambda_b]$ and $\mathcal{J} = \{j : \lambda_j \in [\lambda_a, \lambda_b]\}$. Thus, it might be beneficial to apply a filter to the quantum state $|u\rangle$ to remove all the contributions from eigenvalues that are outside the interval of interest $[\lambda_a, \lambda_b]$, and produce an initial, filtered state $|u_f\rangle$ such that

$$|u_f\rangle \sim \sum_{\lambda_j \in [\lambda_a, \lambda_b]} W_{uj} |\lambda_j\rangle. \tag{30}$$

Quantum eigenvalue filtering has been already studied in the literature [13,54,55] and could find an application in our problem. We note that also for classical algorithms it is possible to restrict the eigenvalue search in a specific range using filter diagonalization methods [56–59].

Needless to say, our algorithm relies on the efficient implementation of the matrix oracles. As discussed in Sec. II C, this usually requires the implementation of reversible, coherent arithmetic that comes at the cost of additional ancilla qubits. Alternatively, one could adapt state preparation techniques that avoid the need of coherent arithmetic [19,60] to this task.[4] In case the matrix elements are already stored in a classical memory, these data would need to be accessed by the oracle using a QRAM [61], which can lead to severe overheads destroying any exponential speedup (see discussion in Refs. [4,62]). For the milling use case discussed in Sec. I B, this would mean that the discretization of the problem would either need to happen in the quantum computer or it must be given in a format that the quantum computer can access efficiently. Rather than an intrinsic limitation, we believe that this is an interesting scientific question and underexplored area of research.

[4]In fact, the oracles are ultimately used to prepare the state in Eq. (B5).

Finally, we point out that our algorithm has some connection with the vector fitting algorithm [63], where the goal is, given K observations of $G_{uu}(\upsilon)$ at $\upsilon = i\omega^{(k)}$, $k \in \{1, \ldots, K\}$, to output the best estimate of the weights W_{uj}^2, usually called residues, and the eigenvalues λ_j. In our case, we want to achieve the same task, but using the data obtained from running QPE multiple times to estimate the weights and the eigenvalues.

III. QUANTUM ALGORITHM

In this section, we detail how the local response function $G_{uu}(\upsilon)$ in Eq. (13) can be obtained using QPE. In particular, we describe the algorithm, and study its scaling, using the standard QPE [7,9], that makes use of the inverse quantum Fourier transform. A similar analysis can also be performed using other versions of QPE that use a single ancilla qubit for the phase register [8,13,64,65] or coherent QPE with fewer qubits [66]. As we have seen in Sec. II A, to obtain the response function, we need to solve the eigenproblem associated with the Hermitian matrix in Eq. (4).

A. Matrix encoding

The algorithm is based on the use of QPE as eigenvalue estimator [1]. In our analysis, we focus on a version of QPE inspired by quantum walks [67,68] and similar to the one of Refs. [15,16]. The basic idea of this method is that if we want to estimate the eigenvalues λ of a Hamiltonian H, we could as well estimate a function $f(\lambda)$ as long as the function is invertible. In particular, in our case instead of performing QPE with unitary e^{iH}, we perform it using a unitary V with eigenvalues $e^{\pm i \arccos(\alpha\lambda)}$ with α a normalization factor. The appeal of this method is that V can be implemented exactly using the standard oracles, which simplifies the error analysis. The full QPE is shown in Fig. 3. In what follows, we briefly describe the method in the language of block-encoding and qubitization [4,10,12–14,69].

Let us assume that we have a unitary U_H acting on $a+n$ qubits that realizes an $(\alpha, a, 0)$-block-encoding of our n-qubit Hamiltonian H. Mathematically, this translates into the requirement

$$\langle 0|^{\otimes a} U_H |0\rangle^{\otimes a} = \alpha H, \tag{31}$$

where a is the number of additional ancilla qubits used for the block-encoding. The eigenstates of H with eigenvalue λ will be denoted by $|\lambda\rangle$ in braket notation. Without loss of generality, we assume that U_H is Hermitian, i.e., $U_H = U_H^\dagger$, which implies $U_H^2 = I_{a+n}$.[5] We define the operator Π as the projector onto the subspace with all ancillas in the zero state

$$\Pi = |0\rangle\langle 0|^{\otimes a} \otimes I_n, \tag{32}$$

where I_n denotes the n-qubit identity.

The crucial property that qubitization uses is that the action of a Hermitian block-encoding U_H on the state $|0\rangle^{\otimes a}|\lambda\rangle$ can be written as (see Sec. 10.4 in Ref. [4])

$$U_H|0\rangle^{\otimes a}|\lambda\rangle = \alpha\lambda|0\rangle^{\otimes a}|\lambda\rangle + \sqrt{1-\alpha^2\lambda^2}|0\lambda^\perp\rangle, \tag{33}$$

where $|0\lambda^\perp\rangle$ is a state such that $(\langle 0|^{\otimes a}\langle\psi|)|0\lambda^\perp\rangle = 0$ $\forall$ n-qubit state $|\psi\rangle$ and $\langle 0\lambda'^\perp|0\lambda^\perp\rangle = \delta_{\lambda\lambda'}$. Additionally,

$$U_H|0\lambda^\perp\rangle = \sqrt{1-\alpha^2\lambda^2}|0\rangle^{\otimes a}|\lambda\rangle - \alpha\lambda|0\lambda^\perp\rangle. \tag{34}$$

Eq. (33) and Eq. (34) imply that $\forall\lambda$ we can identify a qubitlike subspace $\mathcal{H}^{(\lambda)} = \text{span}\{|0\rangle^{\otimes a}|\lambda\rangle, |0\lambda^\perp\rangle\}$, so that the action of U_H onto this subspace can be represented by a 2×2 matrix $U_H^{(\lambda)}$ given by

$$U_H^{(\lambda)} = \begin{pmatrix} \alpha\lambda & \sqrt{1-\alpha^2\lambda^2} \\ \sqrt{1-\alpha^2\lambda^2} & -\alpha\lambda \end{pmatrix}. \tag{35}$$

However, the eigenvalues of this matrix are ± 1, since U_H is a reflection and thus, they do not carry information about the eigenvalues λ of H. To remedy this issue we define the "walk" operator V that we use in the QPE subroutine depicted in Fig. 3 as

$$V = U_H(2\Pi - I_{a+n}). \tag{36}$$

In fact, the operator $2\Pi - I_{a+n}$ also preserves the subspaces $\mathcal{H}^{(\lambda)}$ and its action on $\mathcal{H}^{(\lambda)}$ is simply represented by a Pauli Z matrix:

$$(2\Pi - I_{a+n})^{(\lambda)} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{37}$$

Consequently, the action of V on $\mathcal{H}^{(\lambda)}$ is given by

$$V^{(\lambda)} = U_H^{(\lambda)}(2\Pi - I_{a+n})^{(\lambda)} = \begin{pmatrix} \alpha\lambda & -\sqrt{1-\alpha^2\lambda^2} \\ \sqrt{1-\alpha^2\lambda^2} & \alpha\lambda \end{pmatrix}. \tag{38}$$

The matrix $V^{(\lambda)}$ has eigenvalues

$$\mu_\pm = \alpha\lambda \pm i\sqrt{1-\alpha^2\lambda^2} = e^{\pm i \arccos(\alpha\lambda)}, \tag{39}$$

with corresponding eigenvectors

$$|\mu_\pm\rangle = \frac{1}{\sqrt{2}}(|0\rangle^{\otimes a}|\lambda\rangle \mp i|0\lambda^\perp\rangle). \tag{40}$$

Thus, if we start from a state $|\psi\rangle = \sum_\lambda c_\lambda |0\rangle^{\otimes a}|\lambda\rangle$ and perform "ideal" phase estimation using the operator V, then the system will collapse in either $|\mu_+\rangle$ or $|\mu_-\rangle$ and the value $+\arccos(\alpha\lambda)$ or $-\arccos(\alpha\lambda)$, respectively, can be read out in the phase register. In either case, a subsequent measurement of the ancilla register in the computational basis would give all zeros with probability $1/2$ and thus, prepare the eigenstate $|0\rangle^{\otimes a}|\lambda\rangle$.

The procedure we have just described works for an arbitrary Hermitian block-encoding of U_H. In Appendix B, we show for completeness how such a block-encoding can be realized if we have access to the oracles in Eqs. (15) and (17). In this case, $a = n+2$ and the parameter α is given by

$$\alpha = \frac{1}{s\|H\|_{\max}}, \tag{41}$$

where s denotes the sparsity of the matrix H.

[5] If U_H is not Hermitian, then one can add one ancilla qubit and define the Hermitian unitary $U_H^{(\text{new})} = |+\rangle\langle -| \otimes U_H + |-\rangle\langle +| \otimes U_H^\dagger$, which is a $(\alpha, a+1, 0)$-block-encoding of H.

B. Obtaining the response function

In this section, we describe the full quantum algorithm to obtain the diagonal response function in Eq. (13). As we see from Fig. 3, we have three main registers:

(1) An m-qubit register that, at the end of the circuit, stores the information about the eigenphase;

(2) An n-qubit register that represents the Hilbert space associated with the Hamiltonian H;

(3) An $(a+l)$-qubit ancilla register, which includes the a ancillas for the block-encoding described in Sec. III A as well as l ancilla qubits needed for the implementation of the oracles (see Appendix B).

We start by preparing a state $|u\rangle$ associated with one of the vertices $u \in \mathcal{V}$ in the state register (cf. Fig. 3) by using a product of single-qubit Pauli-X gates

$$\prod_{i=0}^{n-1} X_i^{u_i}|0\rangle^{\otimes n} = |u\rangle = \sum_{j=1}^{N} W_{uj}|\lambda_j\rangle, \tag{42}$$

where X_i denotes the Pauli-X gate on qubit i. This is equivalent to an amplitude encoding of the weights W_{uj} in the basis of the eigenstates $|\lambda_j\rangle$ of H. The choice of u affects the set of weights W_{uj} and with it the probability to collapse into the corresponding eigenstate $|\lambda_j\rangle$ upon measurement in this basis.

Including the a ancilla qubits needed for the block-encoding, we can rewrite the state in Eq. (42) in the eigenbasis of the unitary V given in Eq. (40) as

$$|0\rangle^{\otimes a}|u\rangle = \frac{1}{\sqrt{2}} \sum_{j=1}^{N} W_{uj}(|\mu_+^{(j)}\rangle + |\mu_-^{(j)}\rangle). \tag{43}$$

After the execution of QPE with V as unitary operator controlled by the m qubits of the phase register [7], we get

$$\xrightarrow{\text{QPE}} \frac{1}{\sqrt{2}} \sum_{j=1}^{N} W_{uj} \sum_{x=0}^{M-1} \left(a_{x+}^{(j)}|\mu_+^{(j)}\rangle + a_{x-}^{(j)}|\mu_-^{(j)}\rangle\right)|x\rangle, \tag{44}$$

where $M = 2^m$ and $a_{x\pm}^{(j)}$ is the amplitude applied by the QPE subroutine

$$a_{x\pm}^{(j)} = \frac{1}{M} \sum_{z=0}^{M-1} e^{\frac{2\pi i z}{M}(\pm\varphi_j - x)}, \tag{45}$$

with

$$\varphi_j = \frac{M}{2\pi} \arccos\left(\frac{\lambda_j}{s\|H\|_{\max}}\right). \tag{46}$$

The coefficients $a_{x\pm}^{(j)}$ are peaked around $x = \pm\varphi_j \mod M$, respectively. This tells us that the probability $P(x)$ to measure x in the phase register will peak around $\pm\varphi_j \mod M$ for all $j \in \{1, \ldots, N\}$, and it is given by

$$P(x) = \sum_{j=1}^{N} \frac{W_{uj}^2}{2}(|a_{x+}^{(j)}|^2 + |a_{x-}^{(j)}|^2), \tag{47}$$

where we used the orthonormality of $|\mu_\pm^{(j)}\rangle$.

After sampling multiple times from the phase register, we can obtain an estimate of the probabilities Eq. (47), from which we need to extract the eigenvalues λ_j and the weights W_{uj}^2. Here, we assume that we have sufficient resolution to determine the peaks and their centers. Possible practical methods to determine the eigenvalues are discussed in Refs. [20,65,70], and could be tested in practice for our problem. The center $\tilde{\varphi}_j$ of the peak can be translated into the eigenvalues by inverting Eq. (46). The probability to sample from a certain eigenstate is proportional to the weight W_{uj}^2 of interest. The last can be estimated by adding the probabilities of the $2Q$ values closest to $\tilde{\varphi}_j$,

$$W_{uj}^2 \approx 2 \sum_{q=-Q}^{Q-1} P(\lceil\varphi_j\rceil + q), \tag{48}$$

where we assumed that $\tilde{\varphi}_j \in [\lfloor\varphi_j\rfloor, \lceil\varphi_j\rceil]$. The additional factor of 2 is due to the $1/2$ in Eq. (47). The necessary size of Q is further discussed in Appendix C 1 b. The full routine is gathered in Algorithm 1.

In Appendix D, we discuss a more general method which uses a modified Hadamard test to estimate the nonlocal response function $G_{uv}(\upsilon)$.

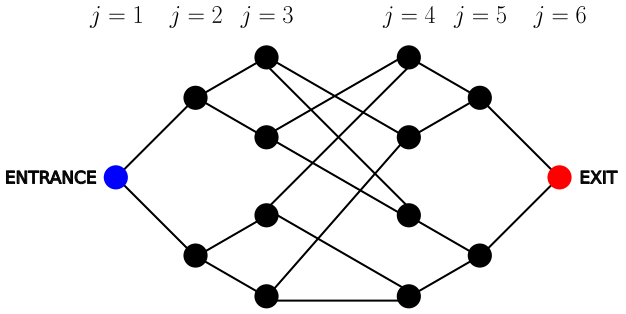


FIG. 4. Example of random glued-trees with $n_c = 3$.

IV. SOLVING THE RANDOM GLUED-TREES PROBLEM

In this section, we show that a straightforward adaptation of the algorithm presented in Sec. III can be used to solve the random glued-trees problem first studied in Ref. [26] in the context of quantum walks. This problem is a well-known example for which quantum algorithms give a provable, oracular exponential speedup compared to classical methods. Recently, Ref. [24] also formulated the problem in terms of coupled harmonic oscillators showing that it can be solved in time polylogarithmic in the problem size, by using a particular encoding and Hamiltonian simulation. We here complement this result by showing that the problem can also be solved in polylogarithmic time using QPE. In what follows, we describe the problem and give our main results, while we provide the details in Appendix E.

The basic setup of the random glued-trees problem is shown in Fig. 4. It consists of a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with two balanced trees each with n_c columns, where the n_cth column of the right tree is glued randomly to the n_cth column of the left tree. In this way, each vertex in the n_cth column of the right tree is connected to exactly two vertices of the n_cth column of the left tree, and vice versa. Let us denote by A the adjacency matrix of a random glued-trees graph. As we see in Fig. 4, we denoted the root of the left tree as the ENTRANCE vertex, and the root of the right tree as the EXIT vertex. Access to

the adjacency matrix A allows us to identify the ENTRANCE and EXIT vertices, since they are the only two vertices with two neighbors. This allows us to define the following random glued-trees problem.

Problem 1. (Random glued-trees problem) Consider a random glued-trees graph with $N = 2(2^{n_c} - 1)$ vertices and n_c the number of columns in each tree. Let A denote its adjacency matrix. Let $\deg(i)$ be the function that given the row i returns the degree of the associated vertex, i.e., the number of vertices connected to it. We assume that the vertices are randomly ordered, but we are given the index of the ENTRANCE vertex, i.e., the row associated with it. Assuming oracle access to the adjacency matrix A and to the function $\deg(i)$ the goal is to find the index of the EXIT vertex.

Note that the function $\deg(i)$ is merely used to identify the ENTRANCE and EXIT vertices, since they are the only vertices of degree 2. Problem 1 has an immediate mapping to a system of coupled oscillators as discussed in Ref. [24]. In fact, we can associate with each vertex $u \in \mathcal{V}$ a unit mass so that the mass matrix is simply the identity matrix I_N. Moreover, we add a unit spring constant to each edge in the graph and an additional unit spring constant to the ENTRANCE and EXIT masses attached to a wall. In this way, according to Eq. (4), the Hamiltonian H coincides with the stiffness matrix and is given by $H = 3I_N - A$. Thus, the eigenvalues and eigenvectors of H are in one-to-one correspondence with those of the adjacency matrix A. In what follows, we assume that we have a block-encoding of A, which is 3-sparse and has $\|A\|_{\max} = 1$.

The mapping of the problem to a quantum computer is the same as the one described in Sec. II B for any coupled oscillator problem. To solve it using QPE, we start by preparing the computational basis string associated with the ENTRANCE vertex (known to us) that we denote as $|\text{ENTRANCE}\rangle$. The idea is now to perform QPE as in Fig. 3, using a block-encoding of the adjacency matrix A. The glued-trees problem enjoys an effective, exponential dimensionality reduction and under the action of the adjacency matrix the evolution of the system is confined to the *column subspace* $\mathcal{H}_{\text{col}}$ (see Appendix E for more details). Following Ref. [26], one can obtain an estimate for the minimum eigenvalue gap Δ_λ of the adjacency matrix projected onto the column subspace A_{col} defined in Eq. (E8):[6]

$$\Delta_\lambda = \frac{8\pi^2}{n_c^3} + \mathcal{O}\left(\frac{1}{n_c^4}\right). \tag{49}$$

Equation (49) holds for any random glued-trees graph. Thus, we see that, despite the fact that the size of the adjacency matrix increases exponentially with the number of columns n_c, i.e., with the number of qubits n needed to encode it, the relevant, minimum gap scales only inverse polynomially. This is exactly one of the features needed for a possible exponential quantum advantage. The idea now is to take the number of qubits m in the phase register large enough to resolve the minimum eigenvalue gap Δ_λ. From Eq. (C8), we require

$$m = \gamma \left\lceil \log_2\left(\frac{3\pi}{\Delta_\lambda}\right)\right\rceil \approx \gamma \left\lceil \log_2\left(\frac{3n_c^3}{8\pi}\right)\right\rceil, \tag{50}$$

with γ a constant, integer factor of our choice such that $\gamma \gg 1$. Accordingly, the number of oracle queries for a single run of QPE will be of order

$$N_{\text{queries}} = \mathcal{O}\left(n_c^3\right). \tag{51}$$

If Eq. (50) is satisfied, then the QPE subroutine described in Sec. III, followed by measurement of the ancilla register with post-selection on the $|0\rangle^{\otimes a}$ outcome, behaves de facto as a measurement in the eigenbasis of A projected onto the column subspace. Thus, after one run of QPE followed by post-selection on $|0\rangle^{\otimes a}$, we have prepared an eigenstate of the adjacency matrix A projected onto the column subspace in the n-qubit register. A final measurement in the computational basis on this register gives a vertex $u \in \mathcal{V}$. We are interested in the probability Prob(EXIT) that this procedure outputs the EXIT vertex. In Appendix E, we obtain an analytical lower bound for this probability [see Eq. (E19)], from which we conclude that

$$\text{Prob(EXIT)} = \mathcal{O}\left(\frac{1}{n_c}\right), \tag{52}$$

which finally proves the overall polynomial scaling of the algorithm in the number of columns n_c, i.e., polylogarithmic in the size of A, complementing the results obtained in Refs. [24,26] with time-evolution-based algorithms.

V. CONCLUSIONS

We have presented a quantum algorithm for obtaining response functions on systems of coupled harmonic oscillators. These kinds of problems also effectively emerge in the study of mechanical, vibrating structures, after discretization, as well as other areas of interest. The particular analytical structure of these functions makes them suitable for a quantum computer. The algorithm is based on the QPE subroutine, that we use as an eigenvalue solver. Importantly, the state preparation at the input of the algorithm just amounts to the preparation of a product state. This allows us to analyze the scaling of the algorithm in terms of space and queries of the matrix oracles. While the algorithm achieves an exponential saving in space, the scaling of the total number of oracles queries is problem-dependent and can be polynomial in the number of qubits only when very specific conditions are met. We identify such a list of sufficient conditions leading to a rigorous worst-case guarantee, but also leave open the possibility for improved practical scaling when a priori information about the structure of the problem is available. In addition, the scaling is mostly limited by statistical error coming from the estimation of the weights of the response functions. Possible improvements of the algorithm can be obtained by combining the algorithm with quantum eigenvalue filtering techniques, as well as quantum amplitude estimation. Finally, we have shown that the algorithm can be adapted to efficiently solve the random-glued trees problems, similarly to other approaches based on Hamiltonian simulation.

[6]Reference [26] gives the result in a slightly different form, namely stating that the eigenvalue spacing is larger than $\frac{2\pi^2}{(1+\sqrt{2})n_c^3} + \mathcal{O}(\frac{1}{n_c^4})$. However, adapting the derivation one can show that Eq. (49) also holds.

While we performed a basic scaling analysis, we believe that a deeper understanding of our algorithm and its potential can be obtained by studying its performances in practice. This could be simply understood via basic classical simulations that aim at obtaining the minimum, relevant eigenvalue gap for some class of oscillator problems. Such analysis would also shed light on the kinds of problems for which large advantages are possible. Additionally, a practical study of the algorithm performances using methods developed for the estimation of multiple eigenvalues [20,65,70] tailored to our problem, would be beneficial. Finally, an open question is whether estimates of the response functions can also be obtained using time-evolution based algorithms for coupled harmonic oscillator as in Ref. [24].

ACKNOWLEDGMENTS

S.D., S.S., and P.K. are funded by the Bundesministerium für Wirtschaft und Klimaschutz (BMWK, Federal Ministry for Economic Affairs and Climate Action) in the quantum computing enhanced service ecosystem for simulation in manufacturing project (QUASIM, Grant No. 01MQ22001A). A.C. acknowledges funding from the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy—Cluster of Excellence Matter and Light for Quantum Computing (ML4Q) Grant No. EXC 2004/1–390534769. A.C. acknowledge funding from the German Federal Ministry of Education and Research (BMBF) in the funding program "Quantum technologies—From basic research to market" (Project QAI2, Contract No. 13N15585). M.B. is supported by the EPSRC Grant No. EP/W032643/1 and the Excellence Cluster—Matter and Light for Quantum Computing (ML4Q).

DATA AVAILABILITY

The algorithm presented in this manuscript were tested for the example of linearly coupled oscillators. For this case, it was implemented and emulated from end-to-end in Qiskit [71], including compilation of the matrix oracles. The corresponding repository containing all the required packages and a demonstration notebook can be found at Ref. [72].

APPENDIX A: ANALYTICAL FORM OF THE RESPONSE FUNCTIONS

In this Appendix, we show that for a system of coupled harmonic oscillators the matrix elements of the response function $G_{uv}(\upsilon)$ in Laplace domain can be written as

$$G_{uv}(\upsilon) = \frac{1}{\sqrt{m_u m_v}} \sum_{j=1}^{N} \frac{W_{uj} W_{vj}}{\lambda_j + \upsilon^2}. \tag{A1}$$

We remark that this is a well-known fact in the study of harmonic systems, in particular in the theory of electrical circuits, where it appears in various forms [73–75]. Also, it is the base for the black-box quantization method for quantum electrical circuits [76]. Here we provide a derivation for our problem adapted from Ref. [77].

Let us consider the case in which we apply to oscillator v an impulse $f_v(t) = \tilde{f}_v \delta(t)$, while no force is applied to the other oscillators, i.e., $f_{v'}(t) = 0$ for $v' \neq v$. In terms of the normal mode variables in Eq. (6), the dynamical equation in time domain Eq. (9) reads

$$\frac{d^2 y_j}{dt^2} = -\lambda_j y_j + \frac{W_{vj}}{\sqrt{m_v}} \tilde{f}_v \delta(t), \tag{A2}$$

which in Laplace domain gives

$$Y_j(\upsilon) = \frac{1}{\sqrt{m_v}} \frac{W_{vj}}{\lambda_j + \upsilon^2} \tilde{f}_v, \tag{A3}$$

for $j \in \{1, \ldots, N\}$. From Eq. (8), we get that the Laplace transform of $x_u(t)$ can be written as

$$X_u(\upsilon) = \frac{1}{\sqrt{m_u}} \sum_{j=1}^{N} W_{uj} Y_j(\upsilon) = \frac{1}{\sqrt{m_u m_v}} \sum_{j=1}^{N} \frac{W_{uj} W_{vj}}{\lambda_j + \upsilon^2} \tilde{f}_v, \tag{A4}$$

from which Eq. (A1) follows.

APPENDIX B: HERMITIAN BLOCK-ENCODING FROM THE MATRIX ORACLES

Here we detail how a Hermitian $(\alpha, a, 0)$-block-encoding U_H of a Hamiltonian H can be constructed using the matrix oracles O_P and O_ϑ defined in Eqs. (15) and (17), respectively. In particular, we limit ourselves to the case when H is real, i.e., symmetric, and all its diagonal elements are positive, which is the case for the matrix in Eq. (4) we are interested in. We start by considering a $2(n+1)$-qubit register, where the first $a = n+2$ qubits will be the ancilla qubits needed for the block-encoding. We take the block-encoding unitary U_H to be of the following form:

$$U_H = U_T^\dagger \mathrm{SWAP}_{n+1} U_T, \tag{B1}$$

where SWAP_{n+1} swaps two $(n+1)$-qubit registers, i.e., $\mathrm{SWAP}_{n+1}|\psi\rangle|\varphi\rangle = |\varphi\rangle|\psi\rangle$ for any $(n+1)$-qubit states $|\psi\rangle, |\varphi\rangle$. Thus, Eqs. (31) and (B1) imply that

$$\alpha H = \langle 0|^{n+2} U_T^\dagger \mathrm{SWAP}_{n+1} U_T |0\rangle^{\otimes n+2}. \tag{B2}$$

We further assume that the map $U_T|0\rangle^{\otimes n+2}$, which maps an n-qubit quantum state to a $2(n+1)$-qubit quantum state can be written as $(N = 2^n)$

$$U_T|0\rangle^{\otimes n+2} = \sum_{u=1}^{N} |\psi_u, 0, u\rangle\langle u|, \tag{B3}$$

where $|\psi_u\rangle$ denotes an $(n+1)$-qubit state to be determined, $|0\rangle$ the single-qubit zero state and $|u\rangle$ an n-qubit computational basis state. Plugging Eq. (B3) into Eq. (B2), we obtain that the states $|\psi_u\rangle$ need to satisfy

$$\alpha H_{uv} = \langle \psi_u, 0, u|0, v, \psi_v\rangle. \tag{B4}$$

A possible solution is to set α as in Eq. (41) and take

$$|\psi_u\rangle = \frac{1}{\sqrt{s}} \sum_{v \in \tilde{N}_G[u]} (i\,\mathrm{sgn}(u-v))^{\Theta(-H_{uv})} \times \left(\sqrt{\frac{|H_{uv}|}{\|H\|_{\max}}} |0\rangle + \sqrt{1 - \frac{|H_{uv}|}{\|H\|_{\max}}} |1\rangle \right) |v\rangle, \tag{B5}$$

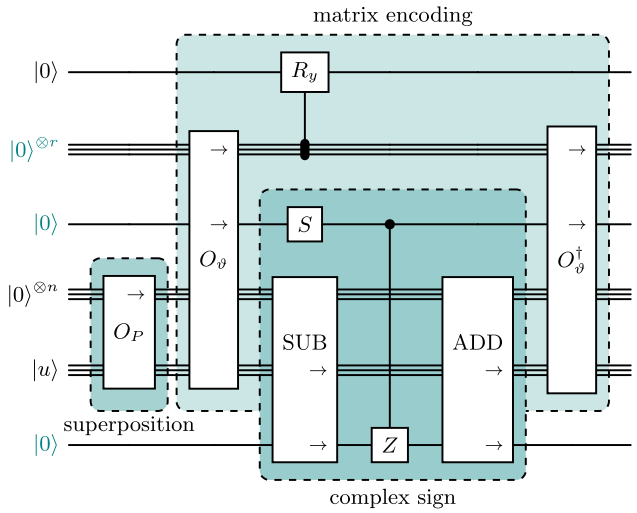


FIG. 5. Circuit for the state transformation U_T. It consists of three parts. The preparation of the superposition with O_P, the matrix encoding in the amplitude and the multiplication with the complex sign. The order of the registers in here is chosen for a compact circuit layout. The first and fourth register combine into $|\psi_u\rangle$ of Eq. (B3). The fifth register is $|u\rangle$ in Eq. (B3). The second, third, and last registers (in blue) are of auxiliary nature and will be reinitialized after the full block-encoding of H. In the second register we store the intermediate angle [Eq. (16)]. The third register is used for the binary sign $\Theta(-H_{uv})$ and the last for $\Theta(\mathrm{sgn}(u-v))$.

where $\mathrm{sgn}(\cdot)$ is the sign function.[7] Note that Eq. (B5) requires the diagonal elements to be nonnegative to give a valid block-encoding. If this is not the case, then one can always enforce this property by adding a suited matrix proportional to the identity to H.

Given the form of the block-encoding in Eq. (B1), all we need to show is how to construct a unitary U_T that satisfies Eq. (B3) with states $|\psi_u\rangle$ given in Eq. (B5). The implementation of U_T consists of three parts. First preparing a superposition of all states $|v\rangle$ with nonzero matrix entries H_{uv}. Second, encoding the matrix entries in the amplitudes, and last, preparing the corresponding signs. All these steps are shown in Fig. 5 as a quantum circuit that involves the matrix oracles O_P and O_ϑ.

We now describe in detail each transformation. As we see from Fig. 5 we have a total of six different registers. The three register highlighted in blue are additional ancilla registers needed for the implementation of the oracles, namely:

(i) an r-qubit ancilla register to store the value of the angles ϑ_{uv} given in Eq. (16);

(ii) an ancilla qubit to store the value of the sign of the matrix elements H_{uv};

(iii) an ancilla qubit (bottom one) to store the value of the sign of $u-v$.

The remaining registers form a $(2n+1)$-qubit register that together with an additional qubit (not shown in the figure) will be used to implement the $2(n+1)$-qubit block-encoding we described at the beginning of this Appendix.

[7]We set $\Theta(0)=0$ and $\mathrm{sgn}(0)=1$.

The system starts in the state $|0\rangle^{\otimes r+2+n}|u\rangle|0\rangle$. After, the application of the position oracle O_P defined in Eq. (15) the state becomes

$$\xrightarrow{O_P} \frac{1}{\sqrt{s}} \sum_{v\in\tilde{N}_{\mathcal{G}}[u]} |0\rangle^{\otimes r+2}|v\rangle|u\rangle|0\rangle, \tag{B6}$$

and after the oracle O_ϑ given in Eq. (17)

$$\xrightarrow{O_\vartheta} \frac{1}{\sqrt{s}} \sum_{v\in\tilde{N}_{\mathcal{G}}[u]} |0\rangle|\vartheta_{uv}, \Theta(-H_{uv})\rangle|v,u\rangle|0\rangle. \tag{B7}$$

We continue by reading the angle ϑ_{uv} in the ancilla register with a series of controlled y-rotations $\mathrm{c}R_y = \prod_{k=1}^{r} \mathrm{c}^l R_y(2^{-k}\pi)$ with target the ancilla qubit on the top of Fig. 5 and control the kth qubit in the binary representation of $|\vartheta_{uv}\rangle$ in Eq. (19).[8] This gives

$$\xrightarrow{\mathrm{c}R_y} \frac{1}{\sqrt{s}} \sum_{v\in\tilde{N}_{\mathcal{G}}[u]} \left(\sqrt{\frac{|H_{uv}|}{\|H\|_{\max}}}|0\rangle + \sqrt{1-\frac{|H_{uv}|}{\|H\|_{\max}}}|1\rangle \right) \otimes |\vartheta_{uv}, \Theta(-H_{uv})\rangle|v,u\rangle|0\rangle. \tag{B8}$$

It remains to implement the coefficient $i\,\mathrm{sgn}(u-v)$ for negative H_{uv}. The phase $i^{\Theta(-H_{uv})}$ can be realized with an S gate to the sign-qubit $|\Theta(-H_{uv})\rangle$:

$$\xrightarrow{S} \frac{1}{\sqrt{s}} \sum_{v\in\tilde{N}_{\mathcal{G}}[u]} i^{\Theta(-H_{uv})} \left(\sqrt{\frac{|H_{uv}|}{\|H\|_{\max}}}|0\rangle + \sqrt{1-\frac{|H_{uv}|}{\|H\|_{\max}}}|1\rangle \right) \otimes |\vartheta_{uv}, \Theta(-H_{uv})\rangle|v,u\rangle|0\rangle. \tag{B9}$$

For the relative sign, we subtract the $|v\rangle$ register from the $|u\rangle$ for which we need to increase the target register by one qubit to store the sign

$$|v\rangle|u,0\rangle \xrightarrow{\mathrm{SUB}} |v\rangle|u-v \mod 2^{n+1}\rangle. \tag{B10}$$

Notice that the most significant bit of $u-v \mod 2^{n+1}$, which is a $(n+1)$-bit number, encodes the sign of the regular subtraction $u-v$, since both u and v are n-bit numbers. Thus, we can write

$$|u-v \mod 2^{n+1}\rangle = |u-v \mod 2^{n}\rangle|\Theta(v-u)\rangle, \tag{B11}$$

where the register associated with $|u-v \mod 2^n\rangle$ is an n-qubit register. The subtraction gives the state

$$\xrightarrow{\mathrm{SUB}} \frac{1}{\sqrt{s}} \sum_{v\in\tilde{N}_{\mathcal{G}}[u]} i^{\Theta(-H_{uv})} \left(\sqrt{\frac{|H_{uv}|}{\|H\|_{\max}}}|0\rangle + \sqrt{1-\frac{|H_{uv}|}{\|H\|_{\max}}}|1\rangle \right) \otimes |\vartheta_{uv}, \Theta(-H_{uv})\rangle|v, u-v \mod 2^n\rangle|\Theta(v-u)\rangle. \tag{B12}$$

A controlled Z gate with control the qubit in the state $|\Theta(-H_{uv})\rangle$ and target the one in the state $|\Theta(v-u)\rangle$ (represented by the qubit at the bottom in Fig. 5) yields the desired

[8]We use the standard definition for the y-rotation in here $R_y(\vartheta) = e^{-i\vartheta\sigma_y/2}$, where σ_y is the second Pauli matrix.

coefficient

$$\xrightarrow{cZ} \frac{1}{\sqrt{s}} \sum_{v\in\tilde{N}_G[u]} (i\mathrm{sgn}(u-v))^{\Theta(-H_{uv})} \times \left(\sqrt{\frac{|H_{uv}|}{\|H\|_{\max}}}|0\rangle + \sqrt{1-\frac{|H_{uv}|}{\|H\|_{\max}}}|1\rangle\right) \otimes |\vartheta_{uv}, \Theta(-H_{uv})\rangle|v, u-v \mod 2^n\rangle|\Theta(v-u)\rangle. \quad \text{(B13)}$$

At last we invert the subtraction SUB using the adder ADD

$$\xrightarrow{\text{ADD}} \frac{1}{\sqrt{s}} \sum_{v\in\tilde{N}_G[u]} (i\mathrm{sgn}(u-v))^{\Theta(-H_{uv})} \times \left(\sqrt{\frac{|H_{uv}|}{\|H\|_{\max}}}|0\rangle + \sqrt{1-\frac{|H_{uv}|}{\|H\|_{\max}}}|1\rangle\right) \otimes |\vartheta_{uv}, \Theta(-H_{uv})\rangle|v, u\rangle|0\rangle, \quad \text{(B14)}$$

and the oracle O_ϑ using $O_\vartheta^\dagger$

$$\xrightarrow{O_\vartheta^\dagger} \frac{1}{\sqrt{s}} \sum_{v\in\tilde{N}_G[u]} (i\mathrm{sgn}(u-v))^{\Theta(-H_{uv})} \times \left(\sqrt{\frac{|H_{uv}|}{\|H\|_{\max}}}|0\rangle + \sqrt{1-\frac{|H_{uv}|}{\|H\|_{\max}}}|1\rangle\right) \otimes |0\rangle^{\otimes r+1}|v, u\rangle|0\rangle, \quad \text{(B15)}$$

and thus, we have effectively prepared the state $|\psi_u\rangle$ in Eq. (B5). Notice that in this procedure the $r+1$ ancilla qubits needed for the implementation of O_ϑ and the ancilla qubit that stores the sign of $u-v$ return to the zero state, and this is also the case for the full block-encoding in Eq. (31). This is why we should not count them among the a ancilla qubits of the block-encoding, which are thus $a = n+2$.

The circuit implementation of U_H in the block-encoding Eq. (B1) is shown in Fig. 6. Note that the total number of qubits was increased by one. This is the last one of the $a = n+2$ ancilla qubits necessary for the Hilbert-space extension. The operator in front is reflecting around $|0\rangle^{\otimes n+2}$ and transforms U_H into the quantum walk operator defined in Eq. (36). It consists of a multi-controlled and a regular Z gate. The QPE demands a controlled version which is why we have another qubit which controls the CNOT gate in the reflection and the SWAP gates in U_H.

Qubit requirements

The full QPE estimation is applied to three registers. The m-qubit phase register, the n-qubit state register and the $(a+l)$-ancilla register used by the controlled walk operator cV. We will focus on the $n+a+l+1$ qubits required for the implementation of cV. For this we start with U_T described in Fig. 5. The total number of qubits n_{U_T} required for the state transformation is given by

$$n_{U_T} = 2n+1+\max(a_P, r+1+a_\vartheta, r+2+a_{\text{ADD}}), \quad \text{(B16)}$$

where $(r+1)$ is the number of qubits used by O_ϑ to store the angle ϑ_{uv} and the sign of H_{uv}. Similar to m it effects the accuracy of the results. Moreover, the number of ancilla qubits necessary for the position and angle oracles, and the quantum adder are given by a_P, a_ϑ and a_{ADD}, respectively. In Eq. (B16), we used that the ancilla qubits used in O_P can be reused in O_ϑ, while the ancilla qubits used there can be used again in O_{ADD}.

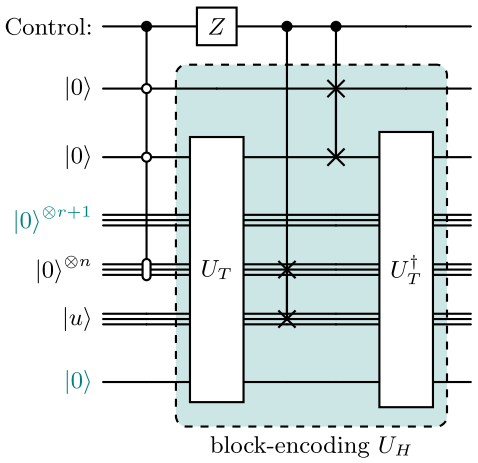


FIG. 6. Controlled V operator with V defined in Eq. (36). The circuit starts with a controlled reflection around $|0\rangle^{\otimes n+2}$ that is implemented by a multi-controlled and a normal Z gate to the control qubit. After this we see the controlled version of the block-encoding $U_H = U_T^\dagger \mathrm{SWAP}_{n+1} U_T$.

The state transformation U_T appears twice in the circuit for cV (cf. Fig. 6). The circuit is further extended by a control qubit and the last qubit necessary for the block-encoding. Hence, we have a total number of n_{cV} qubits necessary for the controlled quantum walk, which is

$$n_{cV} = 2n+3+\max(a_P, r+1+a_\vartheta, r+2+a_{\text{ADD}}, n). \quad \text{(B17)}$$

Here we see additional n ancilla qubits, that are required for the Toffoli gates in the mutli-controlled Z gate in Fig. 6. Those can be reused again. In total we have m repetitions of cV with varying control qubit which leads to the scaling described in Eq. (23). There we replaced m with the scaling Eq. (C34) described in Appendix C.

APPENDIX C: DETAILED ERROR ANALYSIS

In this Appendix, we discuss the effect of discretization and statistical errors on the estimated eigenvalues $\tilde{\lambda}_j$ and weights $\tilde{W}_{uj}^2$ and how one can control the maximum error by increasing the size of the phase register. This analysis proves the statement of Theorem 1.

In particular, we analyze the effect of the discretization of the phase when it is stored as binary number in the m-qubit register with focus on the eigenvalue in Appendix C 1 a and the weights in Appendix C 1 b. The size of the phase register determines the number of oracle queries necessary to implement the controlled-V^{2^k} operations in Algorithm 1 shown in Fig. 3. We discuss this connection in Appendix C 1 c. We point out that the number of oracle queries constitutes the main limitation of the algorithm in most *practical* cases (see discussion in Sec. II D). At last, in Appendix C 2, we describe the connection between the sample size and the statistical

accuracy of the estimated weights. In the following, for notational simplicity, we focus on the nondegenerate systems, but the results carry over to the degenerate case as discussed in Sec. II B.

1. Phase register size for given tolerances

a. Eigenvalue approximation

The following analysis is more easily understood by considering the case in which we perform QPE using the operator V as in Fig. 3, but taking as initial state in the n-qubit register a single eigenstate of H $|\lambda\rangle$. When we measure the m-qubit phase register at the end of QPE we get a bitstring that represents an integer number between 0 and $M-1$ with $M=2^m$. This gives an estimate of $\pm\varphi \mod M \in [0, M)$ with φ [cf. Eq. (46)] given by

$$\varphi = \frac{M}{2\pi}\arccos\left(\frac{\lambda}{s\|H\|_{\max}}\right). \tag{C1}$$

Accordingly, there exists a $\tilde{\varphi} \in \{0, 1, \ldots, 2^{m-1}-1\}$ such that[9]

$$|\varphi - \tilde{\varphi}| \leqslant \tfrac{1}{2}, \tag{C2}$$

to which we can associate an estimated eigenvalue $\tilde{\lambda}$ by inverting Eq. (C1). The phase stored in the ancilla register corresponds to the estimated $\tilde{\lambda}$ as follows:

$$\tilde{\varphi} = \frac{M}{2\pi}\arccos\left(\frac{\tilde{\lambda}}{s\|H\|_{\max}}\right). \tag{C3}$$

Thus, we get

$$\frac{M}{2\pi}\left|\arccos\left(\frac{\lambda}{s\|H\|_{\max}}\right) - \arccos\left(\frac{\tilde{\lambda}}{s\|H\|_{\max}}\right)\right| \leqslant \frac{1}{2}. \tag{C4}$$

The mean value theorem for the arc-cosine function states that for $x > \tilde{x}$, $\exists x_0 \in (\tilde{x}, x)$ such that

$$\frac{|\arccos x - \arccos \tilde{x}|}{|x-\tilde{x}|} = \frac{1}{\sqrt{1-(x_0)^2}}. \tag{C5}$$

Applying the mean value theorem to our case we get that $\exists\lambda_0 \in (\tilde{\lambda}, \lambda)$ if $\lambda > \tilde{\lambda}$ or $\exists\lambda_0 \in (\lambda, \tilde{\lambda})$ if $\lambda < \tilde{\lambda}$, such that

$$\begin{aligned} s\|H\|_{\max}\frac{\left|\arccos\left(\frac{\lambda}{s\|H\|_{\max}}\right) - \arccos\left(\frac{\tilde{\lambda}}{s\|H\|_{\max}}\right)\right|}{|\lambda-\tilde{\lambda}|} \\ = \frac{1}{\sqrt{1-\left(\frac{\lambda_0}{s\|H\|_{\max}}\right)^2}} \geqslant 1, \end{aligned} \tag{C6}$$

where the last inequality holds since $0 \leqslant \lambda_0 \leqslant s\|H\|_{\max}$. Combining Eq. (C4) and Eq. (C6), we get

$$|\lambda - \tilde{\lambda}| \leqslant \frac{\pi s\|H\|_{\max}}{M}, \tag{C7}$$

which tells us that the error of the estimated eigenvalue is below ε if the number of qubits in the phase register satisfies

$$m \geqslant m^{(1)}_{\min} = \left\lceil \log_2\left(\frac{\pi s\|H\|_{\max}}{\varepsilon}\right)\right\rceil. \tag{C8}$$

[9]The first bit in the phase register just represents the sign of $\pm\varphi$ mod M, so φ is represented with $m-1$ bits.

b. Weight approximation

We analyze the error on the estimated weights W^2_{uj} of the local response function defined in Eq. (48) due to the finite size of the phase register. In our analysis we assume that our estimated peak $\tilde{\varphi}_j$ satisfies $\lceil\tilde{\varphi}_j\rceil = \lceil\varphi_j\rceil$ with φ_j associated with the true eigenvalue λ_j as in Eq. (46).[10] Accordingly, our estimator for the weight W^2_{uj} is

$$\tilde{W}^2_{uj} = 2\sum_{q=-Q}^{Q-1} P(\lceil\varphi_j\rceil + q), \tag{C9}$$

with the range Q to be determined.

The parameter Q should be taken large enough to guarantee

$$\left|\tilde{W}^2_{uj} - W^2_{uj}\right| \leqslant \delta. \tag{C10}$$

An increase in Q prevents two things. First, it limits the probability we miss by adding only over the $2Q$ probabilities for values closest to the center of a peak. This is connected to the lower bound of $\tilde{W}^2_{uj} - W^2_{uj}$. The second is that we need to increase the distance between two peaks to fit $2Q$ between them which prevents probability "leakage" between them. This is connected to the upper bound of $\tilde{W}^2_{uj} - W^2_{uj}$.

For the lower limit we need to study the probability to measure x in the phase register defined in Eq. (47) and focus only on one contribution

$$P(x) \geqslant \frac{W^2_{uj}}{2}|a^{(j)}_{x+}|^2. \tag{C11}$$

We will now further develop $|a^{(j)}_{x+}|^2$ by using its definition Eq. (45) and the geometric series

$$\sum_{z=0}^{M-1} x^z = \frac{1-x^M}{1-x}. \tag{C12}$$

Thus, we have

$$\begin{aligned} |a^{(j)}_{x+}|^2 &= \frac{1}{M^2}\left|\frac{1-e^{2\pi i(\varphi_j - x)}}{1-e^{2\pi i(\varphi_j - x)/M}}\right|^2 \\ &= \frac{1}{M^2}\left[\frac{\sin(\pi(\varphi_j - x))}{\sin(\pi(\varphi_j - x)/M)}\right]^2 \geqslant \left[\frac{\sin(\pi(\varphi_j - x))}{\pi(\varphi_j - x)}\right]^2, \end{aligned} \tag{C13}$$

where we used $|\sin x| \leqslant |x|$. Using Eqs. (C9), (C11), and (C13), we get

$$\tilde{W}^2_{uj} \geqslant W^2_{uj}\sum_{q=-Q}^{Q-1}\left[\frac{\sin(\pi(\varphi_j - \lceil\varphi_j\rceil - q))}{\pi(\varphi_j - \lceil\varphi_j\rceil - q)}\right]^2, \tag{C14}$$

which is minimal for $\lceil\varphi_j\rceil - \varphi_j = 1/2$. Hence,

$$\tilde{W}^2_{uj} \geqslant W^2_{uj}\sum_{q=-Q}^{Q-1}\left[\frac{1}{\pi(1/2+q)}\right]^2 = 2W^2_{uj}\sum_{q=1}^{Q}\left[\frac{1}{\pi(q-1/2)}\right]^2. \tag{C15}$$

[10]An analogous derivation is possible if we assume $\lfloor\tilde{\varphi}_j\rfloor = \lfloor\varphi_j\rfloor$.

Now, we introduce the trigamma function

$$\Psi^{(1)}(z) = \sum_{q=0}^{+\infty} \frac{1}{(z+q)^2}, \tag{C16}$$

which for half integer values takes the form

$$\Psi^{(1)}\left(Q+\frac{1}{2}\right) = \frac{\pi^2}{2} - \sum_{q=1}^{Q} \frac{1}{(q-1/2)^2}, \tag{C17}$$

with $Q \in \mathbb{N}$. $\Psi^{(1)}(z)$ satisfies

$$\Psi^{(1)}(z) \leqslant \frac{z+1}{z^2} \leqslant \frac{2}{z}, \quad \forall z > 0. \tag{C18}$$

More details about the trigamma function and a derivation of Eq. (C18) can be found in Ref. [78]. Combining Eqs. (C15), (C17), and (C18) leads to

$$\tilde{W}_{uj}^2 \geqslant W_{uj}^2\left(1 - \frac{2}{\pi^2}\psi^{(1)}\left(Q+\frac{1}{2}\right)\right) \geqslant W_{uj}^2\left(1 - \frac{4}{\pi^2\left(Q+\frac{1}{2}\right)}\right). \tag{C19}$$

Since, $W_{uj}^2 \leqslant 1$, this tells us that our estimation $\tilde{W}_{uj}^2$ is at most δ less than W_{uj}^2 if we choose a Q such that

$$\frac{4}{\pi^2(Q+1/2)} \leqslant \delta, \tag{C20}$$

or

$$Q \geqslant Q_{\min}^{(1)} = \left\lceil \frac{4}{\pi^2\delta} - \frac{1}{2} \right\rceil. \tag{C21}$$

For the upper limit, let us consider

$$\tilde{W}_{uj}^2 - W_{uj}^2 = \sum_{i=0}^{N-1} W_{ui}^2 \sum_{q=-Q}^{Q-1} \left(\left|a_{\lceil\varphi_j\rceil+q,+}^{(i)}\right|^2 + \left|a_{\lceil\varphi_j\rceil+q,-}^{(i)}\right|^2\right) - W_{uj}^2. \tag{C22}$$

Since $\sum_{q=-Q}^{Q-1} |a_{\lceil\varphi_j\rceil+q,+}^{(j)}|^2 \leqslant 1$, we obtain

$$\tilde{W}_{uj}^2 - W_{uj}^2 \leqslant \sum_{i\neq j} W_{ui}^2 \sum_{q=-Q}^{Q-1} \left(\left|a_{\lceil\varphi_j\rceil+q,+}^{(i)}\right|^2 + \left|a_{\lceil\varphi_j\rceil+q,-}^{(i)}\right|^2\right) + W_{uj}^2 \sum_{q=-Q}^{Q-1} \left|a_{\lceil\varphi_j\rceil+q,-}^{(j)}\right|^2. \tag{C23}$$

Let us consider the coefficients $|a_{\lceil\varphi_j\rceil+q,+}^{(i)}|^2$. We can rewrite them as

$$\left|a_{\lceil\varphi_j\rceil+q,+}^{(i)}\right|^2 = \frac{1}{M^2}\left[\frac{\sin(\pi(\gamma_i+\lceil\varphi_i\rceil-\lceil\varphi_j\rceil-q))}{\sin(\pi(\gamma_i+\lceil\varphi_i\rceil-\lceil\varphi_j\rceil-q)/M)}\right]^2, \tag{C24}$$

with $\gamma_i = \varphi_i - \lceil\varphi_i\rceil \in (-1, 0]$. We assume that $|\lceil\varphi_i\rceil - \lceil\varphi_j\rceil > 2Q \ \forall i \neq j$, i.e., that the peaks are separated by at lest $2Q$. Using the fact that $|a_{\lceil\varphi_j\rceil+q,+}^{(i)}|^2$ decreases by increasing $|\lceil\varphi_i\rceil - \lceil\varphi_j\rceil - q|$ for integer values for any fixed $\gamma_i \in (-1, 0]$, we get

$$\left|a_{\lceil\varphi_j\rceil+q,+}^{(i)}\right|^2 \leqslant \frac{1}{M^2}\left[\frac{\sin(\pi(\gamma_i+Q+1))}{\sin(\pi(\gamma_i+Q+1)/M)}\right]^2 = |a_{\gamma_i+Q+1}|^2. \tag{C25}$$

The function $|a_{\gamma+Q+1}|^2$ has exactly one local maximum for $\gamma \in (-1, 0]$, which coincides with the global maximum in the interval. We simply denote by $\bar{\gamma} \in (-1, 0]$ the point where $|a_{\gamma+Q+1}|^2$ is maximal. The same analysis can be repeated for $|a_{\lceil\varphi_j\rceil+q,-}^{(i)}|^2$. Thus, we get

$$\tilde{W}_{uj}^2 - W_{uj}^2 \leqslant \sum_{i\neq j} 2W_{ui}^2 \sum_{q=-Q}^{Q-1} |a_{\bar{\gamma}+Q+1}|^2 + W_{uj}^2 \sum_{q=-Q}^{Q-1} \left|a_{\lceil\varphi_j\rceil+q,-}^{(j)}\right|^2. \tag{C26}$$

We will further assume that $Q < |\pm\varphi_j| < M/2 - Q$, which guarantees that the two peaks φ_j and $-\varphi_j$ are also $2Q$ away from each other and Eq. (C25) is satisfied for $i = j$. The previous condition fails only for $\lambda_j \approx s\|H\|$, where $\pm\varphi_j \approx 0$. This can be prevented by further reducing H with the renormalization factor $\cos(2\pi QM^{-1})$.

Now, we replace the amplitude in the last term of Eq. (C26) as well and have

$$\tilde{W}_{uj}^2 - W_{uj}^2 \leqslant \left(\sum_{i\neq j} 2W_{ui}^2 + W_{uj}^2\right) \sum_{q=-Q}^{Q-1} |a_{\bar{\gamma}+Q+1}|^2. \tag{C27}$$

At last, we replace the amplitudes using Eq. (C13) and use the normalization of the eigenvalues $\sum_i W_{ui}^2 = 1$ to find

$$\tilde{W}_{uj}^2 - W_{uj}^2 \leqslant \frac{4Q}{M^2}\left|\frac{\sin(\pi(\bar{\gamma}+Q+1))}{\sin(\pi(\bar{\gamma}+Q+1)/M)}\right|^2. \tag{C28}$$

Now we use $|\sin x| \leqslant 1$ and $|\sin x| \geqslant |2x/\pi|, \forall |x| \leqslant \pi/2$ and obtain that our estimation is less than δ larger than the exact weight W_{uj}^2 if

$$\tilde{W}_{uj}^2 - W_{uj}^2 \leqslant \frac{Q}{(\bar{\gamma}+Q+1)^2} \leqslant \frac{1}{Q} \leqslant \delta, \tag{C29}$$

if $2Q \leqslant M - 1$. This yields another lower limit for Q, which is

$$Q_{\min}^{(2)} = \left\lceil \frac{1}{\delta} \right\rceil. \tag{C30}$$

A comparison between Eqs. (C21) and (C30) shows that the latter is larger and therefore sufficient to fulfill both conditions. Thus, we set

$$Q \geqslant \left\lceil \frac{1}{\delta} \right\rceil. \tag{C31}$$

The previous analysis depends on the fact that we have a sufficient number of qubits m in the phase register, so that we can distinguish between two neighboring phases. Mathematically, this translates into the condition

$$\min_i |\varphi_i - \varphi_j| \geqslant \min_i \frac{M|\lambda_i - \lambda_j|}{2\pi s\|H\|_{\max}} \geqslant 2Q, \tag{C32}$$

where we used the results obtained in Appendix C 1 a. We can guarantee this if we take the number of qubits in the phase register m such that

$$m \geqslant m_{\min}^{(2)} = \left\lceil \log_2 \frac{4\pi s\|H\|_{\max}}{\delta\Delta_\lambda^{(u)}} \right\rceil, \tag{C33}$$

with $\Delta_\lambda^{(u)}$ is defined in Eq. (22). Combining the results of Eqs. (C8) and (C33), we obtain

$$m \geqslant \max\left(m_{\min}^{(1)}, m_{\min}^{(2)}\right) = \left(\left\lceil \log_2 \frac{\pi s \|H\|_{\max}}{\varepsilon} \right\rceil, \left\lceil \log_2 \frac{4\pi s \|H\|_{\max}}{\delta \Delta_\lambda^{(u)}} \right\rceil\right), \quad \text{(C34)}$$

which justifies Eq. (23) in the main text.

c. Oracle queries for a single run of QPE

The number of qubits m in the phase register determines the number of oracles queries in the algorithm in Fig. 3. As we can see from Fig. 6, the implementation of a controlled V operator requires to use once U_T and once $U_T^\dagger$. The circuit for U_T is shown in Fig. 5 and uses once the position oracle O_P, once the angle oracle O_ϑ and once its inverse $O_\vartheta^\dagger$. Thus, for each U_T we have three oracle calls and a controlled V requires six oracle calls. In order, to perform a single run of QPE with m qubit on the phase register we would need

$$\sum_{k=0}^{m-1} 2^k = 2^m - 1, \quad \text{(C35)}$$

controlled V operations, which yields the number of oracle queries:

$$N_{\text{queries}} = 6(2^m - 1). \quad \text{(C36)}$$

Using the result of Eq. (C34) yields the scaling result in Eq. (25) in the main text.

2. Sample size for given tolerances

We now turn our attention to the scaling of the number of samples N_S. Our aim here is to perform a simplified analysis that should capture the asymptotic behavior of the scaling, while the details will eventually depend on the procedure used to infer eigenvalues and weights from measured data in the phase register (see Ref. [20] for instance). In particular, we work with the assumption that the number of qubits m in the phase register is taken large enough to resolve all the peaks. From Eq. (C8) this means

$$m \gg \left\lceil \log_2 \frac{\pi s \|H\|_{\max}}{\Delta_\lambda} \right\rceil. \quad \text{(C37)}$$

If this holds, then we can unambiguously identify the eigenvalue peaks and neglect possible errors in the assignment of the samples to the eigenvalues. In this regime, QPE de facto behaves as a measurement in the eigenbasis of H, where each eigenvalue λ_j appears with probability $p(j) = W_{uj}^2$.[11] By performing QPE many times, we obtain samples from which we can reconstruct the probability distribution W_{uj}^2.

Let Λ_u be the set of eigenvalues λ_j that have "support" on the oscillator u, i.e.,

$$\Lambda_u = \left\{\lambda_j : W_{uj}^2 > 0\right\}, \quad \text{(C38)}$$

and $N_u = |\Lambda_u|$ the number of elements of Λ_u ($N_u \leqslant N$). W_{uj}^2 is a probability distribution over the finite alphabet Λ_u.

Let $p(\lambda)$ be a generic probability distribution over the finite alphabet Λ_u. With each $\lambda_j \in \Lambda_u$ we can associate a Bernoulli random variable $X_{\lambda_j}(\lambda)$ defined $\forall \lambda \in \Lambda_u$ as

$$X_{\lambda_j}(\lambda) = \begin{cases} 1, & \lambda = \lambda_j, \\ 0, & \lambda \neq \lambda_j. \end{cases} \quad \text{(C39)}$$

Notice that $\mathrm{E}[X_{\lambda_j}] = p(\lambda_j)$. By drawing N_S samples from $p(\lambda)$, we get samples $x_{\lambda_j}^{(i)}$, $i \in \{1, \ldots, N_S\}$ for X_{λ_j}. By taking the arithmetic average of these samples we can construct the empirical estimator for $p(\lambda_j)$:

$$\tilde{p}(\lambda_j) = \frac{1}{N_S} \sum_{i=1}^{N_S} x_{\lambda_j}^{(i)}. \quad \text{(C40)}$$

Our aim is to understand how many samples N_S we have to draw from $p(\lambda)$ to have, with probability $1 - \zeta$, that each $\tilde{p}(\lambda_j)$, $\lambda_j \in \Lambda_u$ estimated $p(\lambda_j)$ with accuracy δ. Mathematically, this means

$$\mathrm{Prob}\left(\bigcup_{\lambda_j \in \Lambda_u} \{|p(\lambda_j) - \tilde{p}(\lambda_j)| \geqslant \delta\}\right) \leqslant \zeta. \quad \text{(C41)}$$

Thus, our aim is to obtain bounds for the left-hand side of Eq. (C41). First, using the union bound we get

$$\mathrm{Prob}\left(\bigcup_{\lambda_j \in \Lambda_u} \{|p(\lambda_j) - \tilde{p}(\lambda_j)| \geqslant \delta\}\right) \leqslant \sum_{\lambda_j \in \Lambda_u} \mathrm{Prob}\left(|p(\lambda_j) - \tilde{p}(\lambda_j)| \geqslant \delta\right). \quad \text{(C42)}$$

Now, an application of Hoeffding's inequality [79,80] to each random variable X_{λ_j} yields

$$\mathrm{Prob}(|p(\lambda_j) - \tilde{p}(\lambda_j)| \geqslant \delta) = \mathrm{Prob}\left(\left|\mathrm{E}[X_{\lambda_j}] - \frac{1}{N_S} \sum_{i=1}^{N_S} x_{\lambda_j}^{(i)}\right| \geqslant \delta\right) \leqslant 2e^{-2N_S\delta^2}. \quad \text{(C43)}$$

Plugging Eq. (C43) into Eq. (C42) and enforcing Eq. (C41) we get

$$\mathrm{Prob}\left(\bigcup_{\lambda_j \in \Lambda_u} \{|p(\lambda_j) - \tilde{p}(\lambda_j)| \geqslant \delta\}\right) \leqslant 2N_u e^{-2N_S\delta^2} \leqslant \zeta, \quad \text{(C44)}$$

which holds if

$$N_S \geqslant \frac{1}{2\delta^2} \ln\left(\frac{2N_u}{\zeta}\right). \quad \text{(C45)}$$

Equation (C45) justifies Eq. (26) in the main text.

APPENDIX D: MODIFICATION FOR NONLOCAL RESPONSE

In this Appendix, we describe how to modify our algorithm to compute nonlocal entries $G_{uv}(\upsilon)$ of the response function whose analytical form is given in Eq. (A1). For the local

[11] In our version of QPE we have two peaks associated with each eigenvalue, but we can simply sum the probability of each peak to get W_{uj}^2.

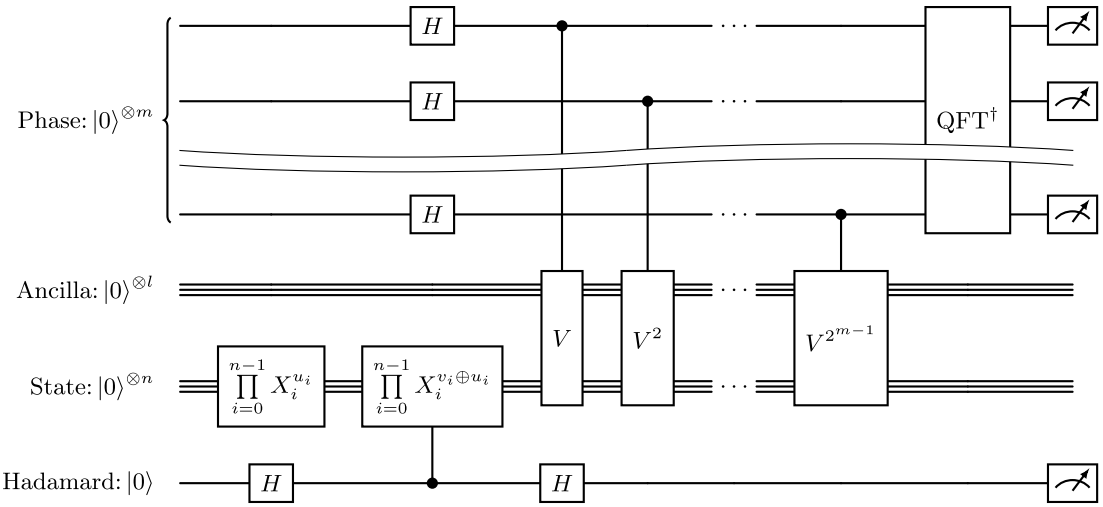


FIG. 7. Modified circuit for the computation of the nonlocal response function. One can see two additions compared to Fig. 3. The first is an additional qubit and the second is a modified Hadamard test which we use to prepare the state $|u\rangle$ next to $|v\rangle$.

contributions the preparation of the computational basis state $|u\rangle$ is crucial to extract the weights W_{uj}^2. The disadvantage of this method is that we can only extract the absolute square of those weights, which does not contain the relative sign and makes this method not suited to evaluate the terms $W_{uj}W_{vj}$ that appear in Eq. (A1).

A way around this is to add another qubit and still prepare $|u\rangle$ in the state register as shown in Fig. 7. We will now use the modified Hadamard test to "add" $|v\rangle$ to this state. At first we apply a Hadamard gate to the additional ancilla qubit starting in $|0\rangle$ (bottom qubit in Fig. 7). Thus, we have

$$\xrightarrow{H} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)|u\rangle. \tag{D1}$$

We now apply a series of CNOT gates with the ancilla qubit as control and the qubits in the n-qubit register as target. The CNOT will be only applied to target qubit i for which $v_i \oplus u_i = 1$. This brings the state $|u\rangle$ to $|v\rangle$ when the control qubit is in $|1\rangle$. Accordingly, after this series of CNOTs we get

$$\xrightarrow{\text{CNOTs}} \frac{1}{\sqrt{2}}(|0\rangle|u\rangle + |1\rangle|v\rangle). \tag{D2}$$

At last we apply another Hadamard gate to the ancilla qubit and obtain

$$\xrightarrow{H} \tfrac{1}{2}(|0\rangle(|u\rangle + |v\rangle) + |1\rangle(|u\rangle - |v\rangle)). \tag{D3}$$

Measuring the additional qubit results in either the sum or difference of those two states

$$\frac{1}{\sqrt{2}}(|u\rangle \pm |v\rangle) = \frac{1}{\sqrt{2}}\sum_{j=1}^{N}(W_{uj} \pm W_{vj})|\lambda_j\rangle. \tag{D4}$$

If we proceed now as before with the local version of the algorithm, depending on the ancilla being in 0 or 1, we would extract the square of those amplitudes

$$\tfrac{1}{2}(W_{uj} \pm W_{vj})^2 = \tfrac{1}{2}W_{uj}^2 + \tfrac{1}{2}W_{vj}^2 \pm W_{uj}W_{vj}, \tag{D5}$$

where we used that W_{uj} and W_{vj} are real. Let $P(0, x)$, $P(1, x)$ the probability of measuring the bitstring x in the phase register given that the ancilla is measured in 0 and 1, respectively. By adapting Eq. (48) with the new amplitudes we find

$$\frac{1}{2}(W_{vj} + W_{uj})^2 \approx 2\sum_{q=-Q}^{Q-1} P(0, \lceil\varphi_j'\rceil + q), \tag{D6a}$$

$$\frac{1}{2}(W_{vj} - W_{uj})^2 \approx 2\sum_{q=-Q}^{Q-1} P(1, \lceil\varphi_j'\rceil + q), \tag{D6b}$$

or, equivalently,

$$W_{vj}W_{uj} \approx \sum_{q=-Q}^{Q-1}(P(0, \lceil\varphi_j'\rceil + q) - P(1, \lceil\varphi_j'\rceil + q)). \tag{D7}$$

APPENDIX E: DETAILS OF THE GLUED-TREES PROBLEM

In this Appendix, we provide the necessary details of the glued-trees problem. We consider a random glued-tree graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ shown in Fig. 4 and described in Sec. IV. We work directly with the quantum formalism so that, with each vertex $u \in \mathcal{V}$, we associate an element in the computational basis $|u\rangle$.

First, we note that, by definition, the action of an adjacency matrix A on a vertex $|u\rangle$ reads

$$A|u\rangle = \sum_{v \in N_{\mathcal{G}}[u]\setminus\{u\}} |v\rangle. \tag{E1}$$

As we see from Fig. 4 the vertices in the glued-trees graph can be organized in $2n_c$ columns. Let us denote by $\text{col}(j)$ the set of vertices in the jth columns starting from the left of the random glued tree graph with $j \in \{1, \ldots, 2n_c\}$. We define the "column" states

$$|\text{col}(j)\rangle = \frac{1}{\sqrt{2^{j-1}}}\sum_{v\in\text{col}(j)} |v\rangle, \quad j \in \{1, \ldots, n_c\}, \tag{E2a}$$

$$|\text{col}(j)\rangle = \frac{1}{\sqrt{2^{2n_c-j}}}\sum_{v\in\text{col}(j)} |v\rangle, \quad j \in \{n_c+1, \ldots, 2n_c\}. \tag{E2b}$$

Notice that ENTRANCE and EXIT vertices correspond to $|\mathrm{col}(1)\rangle$ and $|\mathrm{col}(2n_c)\rangle$, respectively. We denote by $\mathcal{H}_{\mathrm{col}}$ the column subspace:

$$\mathcal{H}_{\mathrm{col}} = \mathrm{span}\{|\mathrm{col}(j)\rangle : j \in \{1, \ldots, 2n_c\}\}. \tag{E3}$$

The projector onto $\mathcal{H}_{\mathrm{col}}$ is

$$\Pi_{\mathrm{col}} = \sum_{j=1}^{2n_c} |\mathrm{col}(j)\rangle\langle\mathrm{col}(j)|. \tag{E4}$$

The crucial fact is that if the system starts in the column subspace, then an application of the adjacency matrix A will give a state in the column subspace. In fact, the action of the adjacency matrix A of the random glued-trees graph on the column states reads

$$A|\mathrm{col}(j)\rangle = \sqrt{2}|\mathrm{col}(j+1)\rangle + \sqrt{2}|\mathrm{col}(j-1)\rangle, \tag{E5}$$

for $j \in \{2, \ldots, n_c - 1\} \cup \{n_c + 1, \ldots 2n_c - 1\}$, and

$$A|\mathrm{col}(1)\rangle = \sqrt{2}|\mathrm{col}(2)\rangle, \tag{E6a}$$

$$A|\mathrm{col}(2n_c)\rangle = \sqrt{2}|\mathrm{col}(2n_c - 1)\rangle, \tag{E6b}$$

$$A|\mathrm{col}(n_c)\rangle = \sqrt{2}|\mathrm{col}(n_c - 1)\rangle + 2|\mathrm{col}(n_c + 1)\rangle, \tag{E6c}$$

$$A|\mathrm{col}(n_c + 1)\rangle = 2|\mathrm{col}(n_c)\rangle + \sqrt{2}|\mathrm{col}(n_c + 2)\rangle. \tag{E6d}$$

For notational simplicity we will denote the column states $|\mathrm{col}(j)\rangle$ as $|j\rangle$ from now on:

$$|\mathrm{col}(j)\rangle \mapsto |j\rangle. \tag{E7}$$

Since in Problem 1 the system starts at the ENTRANCE, i.e., in $|\mathrm{ENTRANCE}\rangle = |j = 1\rangle$, we can effectively restrict to the column subspace and consider the projected adjacency matrix

$$\begin{aligned}\frac{A_{\mathrm{col}}}{\sqrt{2}} = \frac{\Pi_{\mathrm{col}} A \Pi_{\mathrm{col}}}{\sqrt{2}} &= \sum_{j=1}^{n_c-1} (|j\rangle\langle j+1| + \mathrm{H.c.}) \\ &\quad + \sqrt{2}(|n_c\rangle\langle n_c+1| + \mathrm{H.c.}) \\ &\quad + \sum_{j=n_c+1}^{2n_c-1} (|j\rangle\langle j+1| + \mathrm{H.c.}),\end{aligned} \tag{E8}$$

as the effective Hamiltonian of the system. Equation (E8) is the Hamiltonian of a quantum walk on one line with $2n_c$ sites with a "defect" in the middle. Note that from a matrix point of view A_{col} is a tridiagonal symmetric matrix.

As shown in Ref. [26], $A_{\mathrm{col}}/\sqrt{2}$ has eigenvalues of the form

$$\lambda_z = 2\cos z, \tag{E9}$$

where $z = p + iq \in \mathbb{C}$, $p, q \in \mathbb{R}$, satisfies the quantization condition

$$\frac{\sin(z(n_c + 1))}{\sin(zn_c)} = \pm\sqrt{2}. \tag{E10}$$

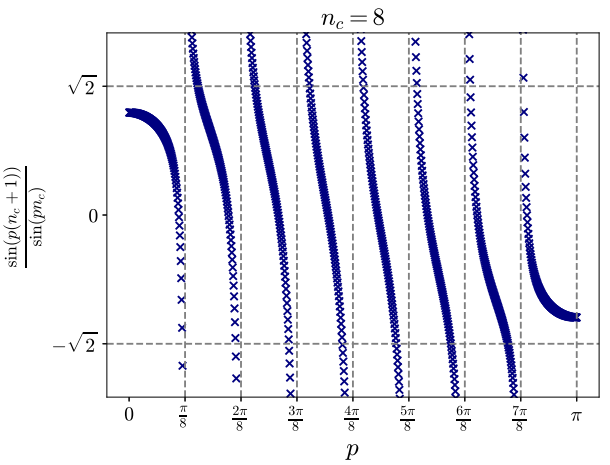


FIG. 8. Quantization condition Eq. (E10) for $n_c = 8$.

The corresponding eigenvector is of the form

$$|\lambda_z\rangle = \frac{1}{\sqrt{\mathcal{N}(z, n_c)}} \left(\sum_{j=1}^{n_c} \sin(zj)|j\rangle \pm \sum_{j=n_c+1}^{2n_c} \sin(z(2n_c + 1 - j))|j\rangle \right), \tag{E11}$$

with $\mathcal{N}(z, n_c)$ a normalization factor. We denote by $\mathcal{Z}$ the set of solutions of Eq. (E10) that are associated with a distinct eigenpair. Thus, $|\mathcal{Z}| = 2n_c$. The quantization condition Eq. (E10) has exactly $2n_c - 2$ real solutions $z = p$ (see Fig. 8 for an example). We denote by $\mathcal{P}$ the set of these solutions. The two remaining eigenpairs can be obtained by setting $z = iq$ and $z = \pi + iq$, but we do not need them to show our result.

In terms of the eigenvectors $|\lambda_z\rangle$, the $|\mathrm{ENTRANCE}\rangle$ state reads

$$|\mathrm{ENTRANCE}\rangle = \sum_{z \in \mathcal{Z}} \frac{\sin z}{\sqrt{\mathcal{N}(z, n_c)}} |\lambda_z\rangle. \tag{E12}$$

We now assume that the number of qubits in the phase register is taken to satisfy Eq. (50) in the main text. With this assumption, the probability of obtaining the eigenstate $|\lambda_z\rangle$ after applying QPE to $|\mathrm{ENTRANCE}\rangle$, as described in Sec. III, and measurement on the ancilla register with post-selection on the $|0\rangle^{\otimes a}$ state, is

$$\mathrm{Prob}(\lambda_z|\mathrm{ENTRANCE}) = \frac{|\sin z|^2}{2\mathcal{N}(z, n_c)}. \tag{E13}$$

Once we obtained one of the $|\lambda_z\rangle$, the probability of finding the EXIT after measurement in the computational basis is

$$\mathrm{Prob}(\mathrm{EXIT}|\lambda_z) = \frac{|\sin z|^2}{\mathcal{N}(z, n_c)}, \tag{E14}$$

Thus, the total probability of finding the EXIT is

$$\mathrm{Prob}(\mathrm{EXIT}) = \sum_{z \in \mathcal{Z}} \frac{|\sin z|^4}{2\mathcal{N}(z, n_c)^2}. \tag{E15}$$

We now want to study analytically the scaling of this probability Prob(EXIT). First, we restrict the sum over the real solutions $z = p$ of Eq. (E10). In this case, the normalization factor $\mathcal{N}(p, n_c)$ reads

$$\mathcal{N}(p,n_c) = \sum_{j=1}^{n_c} \sin^2(pj) + \sum_{j=n_c+1}^{2n_c} \sin^2(p(2n_c+1-j)) = \frac{1}{2}\left[1 + 2n_c - \frac{\sin(p(1+2n_c))}{\sin p}\right] < 2n_c. \quad \text{(E16)}$$

Thus, we get

$$\text{Prob(EXIT)} > \sum_{p\in\mathcal{P}} \frac{\sin^4 p}{2\mathcal{N}(p,n_c)^2} > \frac{1}{8n_c^2} \sum_{p\in\mathcal{P}} \sin^4 p. \quad \text{(E17)}$$

As discussed in Ref. [26] the real solutions concentrate around $\pi\ell/n_c$ with $\ell \in \{1, \ldots, n_c - 1\}$, with corrections of order $\mathcal{O}(1/n_c^2)$. This is also visible in the example shown in Fig. 8 for $n_c = 8$. In particular, we see that there are two solutions around $\pi\ell/n_c$. Thus, for large n_c, we can write

$$\sum_{p\in\mathcal{P}} \sin^4 p = 2\sum_{\ell=1}^{n_c-1} \sin^4\left(\frac{\pi\ell}{n_c}\right) + \mathcal{O}(1) = \frac{3n_c}{4} + \mathcal{O}(1). \quad \text{(E18)}$$

We finally obtain

$$\text{Prob(EXIT)} > \frac{3}{32n_c} + \mathcal{O}\left(\frac{1}{n_c^2}\right), \quad \text{(E19)}$$

which justifies Eq. (52) in the main text.

[1] D. S. Abrams and S. Lloyd, Quantum algorithm providing exponential speed increase for finding eigenvalues and eigenvectors, Phys. Rev. Lett. **83**, 5162 (1999).

[2] B. Bauer, S. Bravyi, M. Motta, and G. K.-L. Chan, Quantum algorithms for quantum chemistry and quantum materials science, Chem. Rev. **120**, 12685 (2020).

[3] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, Quantum computational chemistry, Rev. Mod. Phys. **92**, 015003 (2020).

[4] A. M. Dalzell, S. McArdle, M. Berta, P. Bienias, C.-F. Chen, A. Gilyén, C. T. Hann, M. J. Kastoryano, E. T. Khabiboulline, A. Kubica, G. Salton, S. Wang, and F. G. S. L. Brandão, Quantum algorithms: A survey of applications and end-to-end complexities, arXiv:2310.03011.

[5] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'Brien, A variational eigenvalue solver on a photonic quantum processor, Nat. Commun. **5**, 4213 (2014).

[6] J. Tilly, H. Chen, S. Cao, D. Picozzi, K. Setia, Y. Li, E. Grant, L. Wossnig, I. Rungger, G. H. Booth, and J. Tennyson, The variational quantum eigensolver: A review of methods and best practices, Phys. Rep. **986**, 1 (2022).

[7] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*, 10th ed. (Cambridge University Press, Cambridge, UK, 2011).

[8] A. Y. Kitaev, A. H. Shen, and M. N. Vyalyi, *Classical and Quantum Computation* (American Mathematical Society, Boston, MA, 2002).

[9] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Quantum algorithms revisited, Proc. R. Soc. London A **454**, 339 (1998).

[10] G. H. Low and I. L. Chuang, Hamiltonian simulation by qubitization, Quantum **3**, 163 (2019).

[11] G. H. Low and I. L. Chuang, Optimal Hamiltonian simulation by quantum signal processing, Phys. Rev. Lett. **118**, 010501 (2017).

[12] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics, in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing (STOC'19)* (Association for Computing Machinery, New York, NY, 2019), pp. 193–204.

[13] J. M. Martyn, Z. M. Rossi, A. K. Tan, and I. L. Chuang, Grand unification of quantum algorithms, PRX Quantum **2**, 040203 (2021).

[14] E. Tang and K. Tian, A CS guide to the quantum singular value transformation, arXiv:2302.14324.

[15] D. Poulin, A. Kitaev, D. S. Steiger, M. B. Hastings, and M. Troyer, Quantum algorithm for spectral measurement with a lower gate count, Phys. Rev. Lett. **121**, 010501 (2018).

[16] D. W. Berry, M. Kieferová, A. Scherer, Y. R. Sanders, G. H. Low, N. Wiebe, C. Gidney, and R. Babbush, Improved techniques for preparing eigenstates of fermionic Hamiltonians, npj Quantum Inf. **4**, 22 (2018).

[17] L. Grover and T. Rudolph, Creating superpositions that correspond to efficiently integrable probability distributions, arXiv:quant-ph/0208112.

[18] M. Ozols, M. Roetteler, and J. Roland, Quantum rejection sampling, ACM Trans. Comput. Theory **5**, 1 (2013).

[19] S. McArdle, A. Gilyén, and M. Berta, Quantum state preparation without coherent arithmetic, arXiv:2210.14892.

[20] T. E. O'Brien, B. Tarasinski, and B. M. Terhal, Quantum phase estimation of multiple eigenvalues for small-scale (noisy) experiments, New J. Phys. **21**, 023022 (2019).

[21] A. Dutkiewicz, B. M. Terhal, and T. E. O'Brien, Heisenberg-limited quantum phase estimation of multiple eigenvalues with few control qubits, Quantum **6**, 830 (2022).

[22] Z. Ding and L. Lin, Simultaneous estimation of multiple eigenvalues with short-depth quantum circuit on early fault-tolerant quantum computers, Quantum **7**, 1136 (2023).

[23] R. D. Somma, Quantum eigenvalue estimation via time series analysis, New J. Phys. **21**, 123025 (2019).

[24] R. Babbush, D. W. Berry, R. Kothari, R. D. Somma, and N. Wiebe, Exponential quantum speedup in simulating coupled classical oscillators, Phys. Rev. X **13**, 041041 (2023).

[25] Y. Lee and K. Kanno, Modal analysis on quantum computers via qubitization, arXiv:2307.07478.

[26] A. M. Childs, R. Cleve, E. Deotto, E. Farhi, S. Gutmann, and D. A. Spielman, Exponential algorithmic speedup by a quantum

walk, in *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC'03)* (Association for Computing Machinery, New York, NY, 2003), pp. 59–68.

[27] P. C. S. Costa, S. Jordan, and A. Ostrander, Quantum algorithm for simulating the wave equation, Phys. Rev. A **99**, 012323 (2019).

[28] B. D. Clader, B. C. Jacobs, and C. R. Sprouse, Preconditioned quantum linear system algorithm, Phys. Rev. Lett. **110**, 250504 (2013).

[29] A. Montanaro and S. Pallister, Quantum algorithms and the finite element method, Phys. Rev. A **93**, 032324 (2016).

[30] M. Jung and U. Langer, *Method of Finite Elements for Engineers: An Introduction to the Numerical Foundations and Computer Simulation*, Vol. 2nd revised and extended ed. (Springer Fachmedien Wiesbaden, Wiesbaden, 2013).

[31] L. N. Trefethen and D. Bau, *Numerical Linear Algebra* (SIAM, Philadelphia, PA, 1997).

[32] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed. (Cambridge University Press, New York, NY, 1992).

[33] P. Arbenz, *Lecture Notes on Solving Large Scale Eigenvalue Problems* (ETH Zürich, 2016).

[34] F. Klocke, *Manufacturing Processes 1: Cutting* (Springer, Berlin, Heidelberg, 2011).

[35] S. Maslo, B. Menezes, P. Kienast, P. Ganser, and T. Bergs, Improving dynamic process stability in milling of thin-walled workpieces by optimization of spindle speed based on a linear parameter-varying model, Procedia CIRP **93**, 850 (2020).

[36] J. Munoa, X. Beudaert, Z. Dombovari, Y. Altintas, E. Budak, C. Brecher, and G. Stepan, Chatter suppression techniques in metal cutting, CIRP Annals **65** 785 (2016).

[37] G. Quintana and J. Ciurana, Chatter in machining processes: A review, Int. J. Mach. Tools Manufacture **51**, 363 (2011).

[38] E. Budak, L. Tunç, S. Alan, and H. Nevzat Özgüven, Prediction of workpiece dynamics and its effects on chatter stability in milling, CIRP Annals **61**, 339 (2012).

[39] Y. Altintas, G. Stepan, D. Merdol, and Z. Dombovari, Chatter stability of milling in frequency and discrete time domain, CIRP J. Manufacturing Sci. Technol. **1**, 35 (2008).

[40] S. Schröder, S. Danz, P. Kienast, V. König, P. Ganser, and T. Bergs, An optimization approach for a milling dynamics simulation based on quantum computing, Procedia CIRP **121**, 13 (2024).

[41] V. Rudel, P. Kienast, G. Vinogradov, P. Ganser, and T. Bergs, Cloudbased process design in a digital twin framework with integrated and coupled technology models for blisk milling, Front. Manuf. Technol. **2**, 1021029 (2022).

[42] P. Triverio, S. Grivet-Talocia, M. S. Nakhla, F. G. Canavero, and R. Achar, Stability, causality, and passivity in electrical interconnect models, IEEE Trans. Adv. Packaging **30**, 795 (2007).

[43] C. Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, J. Res. Natl. Bur. Stand. B **45**, 255 (1950).

[44] I. U. OJALVO and M. NEWMAN, Vibration modes of large structures by an automatic matrix-reduction method, AIAA J. **8**, 1234 (1970).

[45] C. C. Paige, The computation of eigenvalues and eigenvectors of very large sparse matrices, Ph.D. thesis, London, 1971.

[46] J. W. Demmel, *Applied Numerical Linear Algebra* (SIAM, Philadelphia, PA, 1997).

[47] T. Ericsson and A. Ruhe, The spectral transformation Lánczos method for the numerical solution of large sparse generalized symmetric eigenvalue problems, Math. Comput. **35**, 1251 (1980).

[48] L. Ruiz-Perez and J. C. Garcia-Escartin, Quantum arithmetic with the quantum fourier transform, Quantum Info. Proc. **16**, 152 (2017).

[49] S. A. Cuccaro, T. G. Draper, S. A. Kutin, and D. Petrie Moulton, A new quantum ripple-carry addition circuit, arXiv:quant-ph/0410184.

[50] T. G. Draper, Addition on a Quantum Computer, arXiv:quant-ph/0008033.

[51] V. Vedral, A. Barenco, and A. Ekert, Quantum networks for elementary arithmetic operations, Phys. Rev. A **54**, 147 (1996).

[52] M. Christandl, N. Datta, A. Ekert, and A. J. Landahl, Perfect state transfer in quantum spin networks, Phys. Rev. Lett. **92**, 187902 (2004).

[53] M. M. Gilles Brassard, Peter Høyer and A. Tapp, Quantum amplitude amplification and estimation, in *Quantum Computation and Information*, edited by J. Samuel J. Lomonaco and H. E. Brandt (Contemporary Mathematics, Singapore, 2002), Vol. 305, pp. 53–74.

[54] L. Lin and Y. Tong, Optimal polynomial based quantum eigenstate filtering with application to solving quantum linear systems, Quantum **4**, 361 (2020).

[55] R. M. Parrish and P. L. McMahon, Quantum filter diagonalization: quantum eigendecomposition without full quantum phase estimation, arXiv:1909.08925.

[56] M. R. Wall and D. Neuhauser, Extraction, through filter-diagonalization, of general quantum eigenvalues or classical normal mode frequencies from a small number of residues or a short-time segment of a signal. I. Theory and application to a quantum-dynamics model, J. Chem. Phys. **102**, 8011 (1995).

[57] R. Chen and H. Guo, A general and efficient filter-diagonalization method without time propagation, J. Chem. Phys. **105**, 1311 (1996).

[58] V. A. Mandelshtam and H. S. Taylor, A low-storage filter diagonalization method for quantum eigenenergy calculation or for spectral analysis of time signals, J. Chem. Phys. **106**, 5085 (1997).

[59] T. Ikegami, T. Sakurai, and U. Nagashima, A filter diagonalization for generalized eigenvalue problems based on the Sakurai–Sugiura projection method, J. Comput. Appl. Math. **233**, 1927 (2010).

[60] Y. R. Sanders, G. H. Low, A. Scherer, and D. W. Berry, Black-box quantum state preparation without arithmetic, Phys. Rev. Lett. **122**, 020502 (2019).

[61] V. Giovannetti, S. Lloyd, and L. Maccone, Quantum random access memory, Phys. Rev. Lett. **100**, 160501 (2008).

[62] S. Aaronson, Read the fine print, Nat. Phys. **11**, 291 (2015).

[63] B. Gustavsen and A. Semlyen, Rational approximation of frequency domain responses by vector fitting, IEEE Trans. Power Delivery **14**, 1052 (1999).

[64] A. Y. Kitaev, Quantum measurements and the Abelian Stabilizer Problem, arXiv:quant-ph/9511026.

[65] K. M. Svore, M. B. Hastings, and M. Freedman, Faster phase estimation, Quantum Inf. Comput. **14**, 306 (2014).
[66] P. Rall, Faster coherent quantum algorithms for phase, energy, and amplitude estimation, Quantum **5**, 566 (2021).
[67] A. M. Childs, On the relationship between continuous-and discrete-time quantum walk, Commun. Math. Phys. **294**, 581 (2010).
[68] D. W. Berry and A. M. Childs, Black-box Hamiltonian simulation and unitary implementation, Quantum Inf. Comput. **12**, 29 (2012).
[69] L. Lin, Lecture notes on quantum algorithms for scientific computation, arXiv:2201.08309.
[70] N. Wiebe and C. Granade, Efficient Bayesian phase estimation, Phys. Rev. Lett. **117**, 010503 (2016).
[71] M. Treinish, "Qiskit/qiskit-metapackage: Qiskit 0.44.0 (0.44.0)", Zenodo (2023), https://doi.org/10.5281/zenodo.8190968.
[72] https://go.fzj.de/resp-QPE
[73] R. Newcomb, *Linear Multiport Synthesis*, McGraw-Hill Electronic Sciences Series (McGraw-Hill, New York, NY, 1966).
[74] P. Russer, *Electromagnetics, Microwave Circuit and Antenna Design for Communications Engineering* (Artech House, Boston, MA, 2006).
[75] R. M. Foster, A reactance theorem, Bell Syst. Tech. J. **3**, 259 (1924).
[76] S. E. Nigg, H. Paik, B. Vlastakis, G. Kirchmair, S. Shankar, L. Frunzio, M. H. Devoret, R. J. Schoelkopf, and S. M. Girvin, Black-box superconducting circuit quantization, Phys. Rev. Lett. **108**, 240502 (2012).
[77] A. Ciani, D. P. DiVincenzo, and B. M. Terhal, *Lecture Notes on Quantum Electrical Circuits* (TU Delft OPEN Publishing, Netherlands, 2024).
[78] F. Qi, S. Guo, and B.-N. Guo, Complete monotonicity of some functions involving polygamma functions, J. Comput. Appl. Math. **233**, 2149 (2010).
[79] W. Hoeffding, Probability inequalities for sums of bounded random variables, J. Am. Stat. Assoc. **58**, 13 (1963).
[80] A. Dembo and O. Zeitouni, *Large Deviations Techniques and Applications*, 2nd ed., Applications of Mathematics (Springer, New York, NY, 1998).